2012

Statistical methods for identifying differentially expressed genes using hierarchical models

Steven Peder Lund
Iowa State University

Statistical methods for identifying differentially expressed genes using hierarchical models

by

Steven Peder Lund

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Statistics

Program of Study Committee:

Dan Nettleton, Major Professor

Gwyn Beattie

Alicia Carriquiry

Kenneth Koehler

Dan Nordman

Iowa State University

Ames, Iowa

2012

DEDICATION

I dedicate this dissertation to my wife Jessica, without whose support I would not have been able to complete this work, and to my children Eliana, Nadia and Nathaniel, who are the source of my motivation.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I would like to thank Dr. Nettleton for his guidance and encouragement. His calm and supportive demeanor throughout the highs and lows of research has made my experience in graduate school much more enjoyable. I feel lucky for receiving the opportunity to work with him and am certain that doing so has made me a much better statistician.

I would also like to Dr. Vardeman for his helpful suggestions and honest assessments during a semester as my mentor in the Preparing Future Faculty program. Dr. Vardeman was instrumental in my receiving the opportunity and making the decision to begin my career working at the National Institute of Standards and Technology.

I would additionally like to thank my parents Tom and Jill for teaching me the importance of education and for providing a wonderful start and a world of opportunity.

CHAPTER 1. Introduction

This dissertation focuses on methods for identifying differentially expressed genes from microarray and RNA-sequence data. Studies focusing on differential expression seek to better understand the complex interactions and response that living organisms have to their environment.

1.1 Gene Expression

Every living cell contains DNA, which is made up of thousands of genes. Genes can be thought of as recipes for constructing proteins, which perform important biological functions, like converting food into energy or controlling which substances can pass through a cell membrane. DNA is made up of four nucleotides, which can be identified by their base. The four bases are adenine (A), cytosine (C), guanine (G), and thymine (T). These bases pair together, creating two strands that twist together forming the double helix shape. Due to the chemical makeup of nucleotides, A bases always pair with T bases and G bases always pairs with C bases. We may then define the complement of a given single strand nucleotide sequence to be the sequence of its pairs. For example, the complement of the sequence ACCGTA is given by TGGCAT. During gene expression, DNA is transcribed into messenger RNA (mRNA). mRNA is similar to a single strand of DNA where T nucleotides have been replaced with uracil (U). Transcribed mRNA contains series of three-nucleotide sequences called codons, each of which corresponds to an amino acid. Proteins are formed during a process called translation, in which amino acids are strung together according to mRNA codons.

1.2 Differential Expression

Understanding the role or function of genes and their proteins is an area of great scientific interest. Examining how gene expression rates are affected by changes in environment is one method researchers

use to understand gene function. Since mRNA is a precursor of protein production, the concentration of a particular mRNA sequence provides information about how frequently the corresponding gene is transcribed. Microarray and RNA-sequence experiments can simultaneously quantify the expression level of mRNA in a sample for thousands of genes, typically collecting expression data from samples in two or more different conditions (i.e. cancerous tissue vs. non-cancerous tissue, control environment vs. water deprived or NaCl amended environment).

When discussing this area of research, change in the average expression level of a single gene across conditions is referred to as differential expression. Differential expression studies do not focus on differences between expressions levels of multiple genes within a sample. Various factors such as gene length, binding affinity, mRNA degradation rates, and biases introduced by sample preparation inhibit the comparison of expression levels between two genes. A gene for which the average expression level varies (is constant) across conditions is said to be differentially (equivalently) expressed. This dissertation describes novel methods for detecting differentially expressed genes from data collected from microarray or RNA-seq experiments.

1.3 Microarray Experiments

Microarray experiments quantify mRNA expression using pre-constructed slides made up of thousands of spots. Each spot is filled with thousands of probes, which are synthetically constructed nucleotide sequences designed to target a specific gene using the complementary binding nature of nucleotides. In a microarray experiment, mRNA is extracted from a sample and converted to complementary DNA (cDNA). A fluorescent dye is attached to nucleotides in the cDNA, which is then applied to the slide where probes bind with cDNA fragments from their corresponding gene in a process called hybridization. Some probes may cross-hybridize (bind with cDNA fragments from genes other than their intended gene), and cDNA fragments that do not bind to a probe are removed. A laser is used to excite the fluorescent dye, causing the spots to emit light. The light intensity from each spot is recorded and serves as an initial measurement of the expression level for its corresponding gene. The raw light intensities undergo a heavily involved process of normalization and background correction, which is a large area of ongoing research that is not the focus of this dissertation. Two-color oligonucleotide

microarray experiments hybridize cDNA from two samples onto each slide, one labeled with green dye and the other with red. Chapter 2 discusses microarray analysis methods that primarily focus on data collected from experiments in which cDNA from a single sample is hybridized to each slide (often manufactured by Affymetrix or Illumina).

There are drawbacks to using microarray technology to examine gene expression. Probe sequences used to fill the spots when constructing a microarray slide must be determined in advance, which means the transcriptome must be well documented before the experiment, and expression levels of genes without a spot on the slide are not measured. From this standpoint, microarrays are not well suited for the discovery of new genes or isoforms. Also, each spot contains a finite number of probes, which can cause a saturation effect making it difficult to measure expression levels of highly expressed genes.

1.4 RNA Sequencing Experiments

Next-Generation Sequencing (NGS) is an ultra-high-throughput technology to determine DNA sequences. One facet of NGS is RNA sequencing (RNA-seq), which can provide discrete count data serving as measures of mRNA expression levels through the following procedure. Messenger RNA is isolated from sample cells, fragmented, and copied to cDNA. The cDNA fragments are then amplified and sequenced, producing strings of nucleotides called reads. The resulting reads are aligned with a reference genome, and the number of reads mapped within each reference gene provide the RNA-seq count data.

Transcription is the process of creating an RNA copy of a sequence of DNA. When discussing this area of research, mRNA nucleotide sequences in the sample are called transcripts. Some genes may incur alternative splicing, causing variations in their transcribed mRNA sequences called isoforms. Some experiments examine the expression of each isoform separately, in which case each gene can potentially have multiple transcripts. A nucleotide sequence resulting from the sequencing of a cDNA fragment that matches a portion of a transcript is called a read. The collection of all reads produced by a single sample (or experimental unit) is called a library. The number of reads contained in a library is referred to as library size. Chapter 3 focuses on methods for analyzing counts of reads that align with a single reference gene. Chapter 4 extends these methods to handle reads that align with multiple

reference transcripts, called multireads. Because isoforms originating from a single gene are often very similar, multireads are particularly prominent in experiments where each isoform is considered as its own transcript. For this reason, Chapter 4 refers to expression levels of transcripts, rather than genes.

At least in principle, RNA-seq count data is a more direct method to quantifying the amount of mRNA produced by a gene than using the fluorescence measures produced with microarray technology. RNA-seq will measure any transcript in a sample, which means there is no need to identify probes prior to measurement or to build a microarray slide. RNA-seq provides data at the resolution of a single nucleotide. These detailed reads provide information about the transcript sequences themselves, in addition to their abundance. Sequence information helps identify allele specific expression and forms of sequence variation like alternative splicing and single nucleotide polymorphisms (SNPs). RNA-seq reads allow one to separately measure the expression of different but very similar transcripts, such as isoforms, that would be difficult to separately measure with microarray technology due to cross hybridization. A drawback of RNA-seq technology is that its distribution and sources of variability currently are not as well understood as are those for microarray experiments.

1.5 General Overview of Recurring Topics

For both RNA-seq and microarray experiments, it is common to collect expression data for thousands of genes with few samples or replicates for each gene. The small number of replicates for each gene limits the power standard one-gene-at-a-time analyses have to detect differential expression. Many analysis methods use hierarchical models to share information across genes when estimating model parameters (variance parameters, in particular), offering substantially improved power and error-rate control. Sharing information across genes is a recurring topic throughout the chapters of this dissertation.

Sharing information across genes can improve accuracy and reduce variability of parameter estimates. However, even after sharing information across genes, estimators are still subject to some non-negligible combination of bias and variability. Some methods assume all genes share a common value of some model parameter and obtain an estimator, informed by thousands of genes, with negligible uncertainty. However, in nearly all cases, assumptions of constant parameters across genes can be shown to be inaccurate. Proceeding with a common parameter estimate for all genes then imposes

a non-negligible bias for some, if not most, genes and adversely affects the detection of differential expression.

In many cases, gene-specific estimators have been developed for parameters originally assumed to be constant across genes. However, the resulting method extensions often fail to account for uncertainty in the gene-specific parameter estimates, which leads to liberal results (i.e. artificially small p-values). Such methods can cause researchers who use them to overestimate the significance of detected differential expression, which can adversely affect gene selection for follow-up studies and inhibit the understanding of gene function. Adequately accounting for uncertainty in parameter estimators, particularly variance parameters that are allowed to vary from gene-to-gene, is another recurring topic in the following chapters.

CHAPTER 2. THE IMPORTANCE OF DISTINCT MODELING STRATEGIES FOR GENE AND GENE-SPECIFIC TREATMENT EFFECTS IN HIERARCHICAL MODELS FOR MICROARRAY DATA

A paper accepted by *Annals of Applied Statistics*

Steven P. Lund and Dan Nettleton

Abstract

When analyzing microarray data, hierarchical models are often used to share information across genes when estimating means and variances or identifying differential expression. Many methods utilize some form of the two-level hierarchical model structure suggested by Kendziorski et al. (2003) in which the first level describes the distribution of latent mean expression levels among genes and among differentially expressed (DE) treatments within a gene. The second level describes the conditional distribution, given a latent mean, of repeated observations for a single gene and treatment. Many of these models, including those used in Kendziorski et al. (2003)'s EBarrays package, assume that expression level changes due to treatment effects have the same distribution as expression level changes from gene to gene. We present empirical evidence that this assumption is often inadequate and propose three-level hierarchical models as extensions to the two-level log-normal based EBarrays models to address this inadequacy. We demonstrate that use of our three-level models dramatically changes analysis results for a variety of microarray data sets and verify the validity and improved performance of our suggested method in a series of simulation studies. We also illustrate the importance of accounting for the uncertainty of gene-specific error variance estimates when using hierarchical models to identify differentially expressed genes.

2.1 Introduction

There are many analytic methods for microarray data that utilize a hierarchical model to share information across genes when estimating mean expression levels. A large subset of these methods model differences in expression levels from gene to gene and differences in expression levels caused by treatment effects with a single distribution. Canonical examples of such methods are implemented in the EBarrays package for R developed by Kendziorski et al. (2003). This work has been influential as indicated by a variety of recent methods that cite Kendziorski et al. (2003) and follow their modeling strategy. Examples include Newton et al. (2004); Yuan and Kendziorski (2006a); Yuan (2006); Yuan and Kendziorski (2006b); Lo and Gottardo (2007); Keles (2007); Wei and Li (2007, 2008); Wu et al. (2007); Jensen et al. (2009); and Rossell (2009).

The analytic methods provided in EBarrays are based on two-level hierarchical parametric models that can be used analyze data from experiments with more than two treatment groups and produce posterior expression pattern probabilities, which can be used to assess the significance of and classify differential expression of genes. The first level of the hierarchical model describes the distribution of latent mean expression levels among genes and among differentially expressed (DE) treatments within a gene. The second level describes the conditional distribution, given a latent mean, of repeated observations for a single gene and treatment.

A necessary user input to models like those included in EBarrays is a list of possible expression patterns. In a two-treatment experiment, the only two expression patterns are equivalent expression and differential expression. In general, each pattern describes how to partition the experimental units into groups based on the experimental conditions or treatments associated with the experimental units. An analysis based on these models can yield a gene-specific posterior probability estimate for each pattern.

The application of hierarchical models to microarray data has many benefits: "sharing" information across genes compensates for having few replicates, users may define expression patterns of interest involving two or more experimental conditions, posterior probabilities assigned to expression patterns are easy to interpret and allow for easy classification or ranking, and simultaneous analysis of all genes in a data set greatly reduces the dimensionality of the inference problem. While the work of Kendziorski et al. (2003) lays a foundation for a powerful method of microarray analysis upon which many methods

have been developed, there is room to relax assumptions and to improve the models described.

The main point of this paper concerns the assumption – implied by the modeling strategy of Kendziorski et al. (2003) – that expression changes across genes have the same distribution as expression changes caused by treatment effects. This assumption is convenient for computational reasons but has undesirable consequences. In particular, if expression differences from gene to gene tend to be larger than treatment effects, the power to identify differentially expressed genes will be reduced. Based on our experience with microarray data, we see no reason to believe that expression differences across genes have the same distribution as expression differences caused by treatment effects in all experiments. Thus, we propose to relax this assumption by adding an additional level to the hierarchy of Kendziorski et al. (2003)'s lognormal-normal (LNN) model. This creates a three-level hierarchical model that we will call the lognormal-normal-normal (LN^3) model.

A secondary point of this paper concerns the assumption of a constant coefficient of variation used in Kendziorski et al. (2003)'s gamma-gamma (GG) and lognormal-normal (LNN) models, which for the latter model, implies an error variance of log expression values that is common to all genes. This assumption is now widely regarded as untenable. To address this, Lo and Gottardo (2007) introduced a method to relax the assumption of the GG and LNN models, and many methods to estimate gene-specific error variances for microarray data have been developed. (See, for example, Baldi and Long (2001); Lonnstedt and Speed (2002); Wright and Simon (2003); Cui et al. (2005).) Kendziorski et al. (2003)'s EBarrays package includes the LNN-moderated variance (LNNMV) method, which uses shrunken point estimates of gene-specific error variances similar to those described by Smyth (2004). We briefly demonstrate that using point estimates of gene-specific error variances without accounting for their uncertainty produces liberal posterior pattern probability estimates, which causes underestimation of the proportion of false positives on a list of significant genes. We propose a simple adaptation to the LNNMV method to account for the uncertainty in gene-specific variance estimates and demonstrate this corrects the liberal bias in the estimated expression pattern posterior probabilities. Finally, we combine our proposed three-level hierarchical modeling strategy with gene-specific error variance modeling to obtain a more general model denoted LN^3MV.

We formally describe the four lognormal based models (LNN, LNNMV, LN^3, and LN^3MV) and corresponding analytic methods in Section 2.2. In Section 2.3, we present empirical evidence from

two example microarray data sets that clearly supports our proposed three-level hierarchical modeling strategy. In Section 2.4, we demonstrate the practical impact of our suggested adaptations by analyzing data from the two microarray experiments with several methods. Section 2.5 describes a variety of simulation studies used to verify the validity and improved power of our suggested methods. For both real and simulated data sets, the use of our proposed three-level hierarchal model dramatically increases power to detect DE genes.

2.2 Model Descriptions

Throughout this paper, we will use the term "group" to denote a set of equivalently expressed (EE) observations from a single gene. Consider a microarray data set with expression values for J genes from each of I experimental units divided among 2 experimental conditions. If for gene j there is no difference between the expression distributions for experimental units under conditions 1 and 2, then the entire set of I observations forms a single group. If for gene j there is a difference between the expression distributions for experimental units under conditions 1 and 2, then the set of observations from experimental units under condition 1 forms one group and the set of observations from experimental units under condition 2 forms a second group. In general, there is at least one group for every gene and at most one group for every combination of gene and experimental condition.

Throughout this section, we will use $G_p(i)$ to denote the group (subset of EE observations) to which the ith experimental unit belongs under the pth expression pattern. For example, suppose there is an experiment with 6 experimental units distributed across 3 treatment groups labeled control, A, and B. If a researcher aims to compare each of treatments A and B to the control, then expression patterns of interest for each gene are p=1: control=A, control=B; p=2: control$\neq$A, control=B; p=3: control=A, control$\neq$B; and p=4: control$\neq$A, control$\neq$B. If experimental units 1 and 2 received the control treatment, 3 and 4 received treatment A, and 5 and 6 received treatment B; then $G_1(i)$=1 for i=1, ..., 6; $G_2(i)$=1 for i=1,2,5,6 and 2 for i=3,4; $G_3(i)$=1 for i=1,2,3,4 and 2 for i=5,6; $G_4(i)$=i/2 rounded up to the nearest integer for all i. We will use P to denote the number of expression patterns of interest and n_p to denote the number of groups under expression pattern p. In the example above, P=4, n_1=1, n_2=n_3=2, and n_4=3.

In each model, the marginal density for $\mathbf{y}_j = (y_{j1}, y_{j2}, \ldots, y_{jI})'$, the vector of observations from the jth gene for I experimental units, is given by $f(\mathbf{y}_j|\boldsymbol{\theta}, \boldsymbol{\pi}) = \sum_{p=1}^{P} \pi_p f_p(\mathbf{y}_j|\boldsymbol{\theta})$, where $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_P)'$, π_p is the probability that a gene follows expression pattern p, $\boldsymbol{\theta}$ is a vector of hyperparameters for the given model, and $f_p(\mathbf{y}_j|\boldsymbol{\theta})$ is the density of $\mathbf{y}_j$ under pattern p according to the given model. The marginal likelihood of the entire data set is given by $\prod_{j=1}^{J} f(\mathbf{y}_j|\boldsymbol{\theta}, \boldsymbol{\pi})$, since observations between genes are considered independent under each of the discussed models. The posterior probability gene j follows expression pattern p given $\mathbf{y}_j$ is $\frac{\pi_p f_p(\mathbf{y}_j|\boldsymbol{\theta})}{\sum_{p=1}^{P} \pi_p f_p(\mathbf{y}_j|\boldsymbol{\theta})}$.

For each model, estimates of $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$ that maximize the marginal likelihood can be obtained using the EM algorithm, treating expression pattern as the unknown variable. When used, gene-specific error variances are estimated and treated as fixed before using the EM algorithm to estimate other model parameters. Marginal densities and posterior probabilities are estimated by treating parameter estimates as the true parameter values in the formulas above.

In the following subsections, we formally define four models and seven methods of analysis. The distinguishing features of the seven methods are summarized in Table 2.1 for future reference.

Table 2.1 Legend for method and model acronyms.

Method	Model	Relies on distinct modeling strategies for differences across genes and differences across DE treatments	Uses gene-specific error variance estimates	Accounts for uncertainty in error variance estimators
LNN	LNN			
LNNMV	LNNMV		✔	
LNNMV*	LNNMV		✔	
LNNGV	LNNMV		✔	✔
LN^3	LN^3	✔		
LN^3MV*	LN^3MV	✔	✔	
LN^3GV	LN^3MV	✔	✔	✔

The methods with acronyms ending in MV* use point estimates of error variances that account for the degrees of freedom used when estimating treatment means (see Section 2.2.3).

2.2.1 The Lognormal-Normal Model

The LNN model for the log scale observation for the jth gene from the ith experimental unit under expression pattern p can be written as

$$y_{ji} = \mu + \tau_{jG_p(i)} + \varepsilon_{ji} \text{ where } \tau_{j1}, \ldots, \tau_{jn_p} \overset{iid}{\sim} N(0, \sigma_\tau^2)$$
$$\text{and } \varepsilon_{j1}, \ldots, \varepsilon_{jI} \overset{iid}{\sim} N(0, \sigma^2)$$

In this expression, μ represents the average expression of all genes and groups, $\tau_{jG_p(i)}$ represents a random group effect for observations from the $G_p(i)$th group (under pattern p) in the jth gene, and ε_{ji} represents a random error.

Under this model, $f_p(\mathbf{y}_j|\boldsymbol{\theta})$ is the density from a multivariate normal distribution with mean vector $(\mu, \ldots, \mu)'$ and pattern specific covariance matrix $\Sigma_p = \sigma^2 I + \sigma_\tau^2 M_p$ where I is the identity matrix and M_p is a symmetric matrix with element $[i, j] = 1$ if experimental units i and j are in the same group under pattern p and $[i, j] = 0$ if experimental units i and j are in different groups. This model has hyperparameters $\boldsymbol{\theta} = (\mu, \sigma^2, \sigma_\tau^2)$.

2.2.2 The Lognormal-Normal-Normal Model

To explicitly model gene effects separately from treatment effects, we propose a three-level hierarchical model, which we denote LN^3. Under the LN^3 model, the log scale observation from the jth gene and the ith experimental unit under expression pattern p is modeled as

$$y_{ji} = \mu + \gamma_j + \tau_{jG_p(i)} + \varepsilon_{ji} \text{ where } \gamma_j \overset{iid}{\sim} N(0, \sigma_\gamma^2),$$
$$\tau_{j1}, \ldots, \tau_{jn_p} \overset{iid}{\sim} N(0, \sigma_\tau^2), \text{ and } \varepsilon_{j1}, \ldots, \varepsilon_{jI} \overset{iid}{\sim} N(0, \sigma^2)$$

In this expression, μ represents the average expression of all genes and groups, γ_j represents a random gene effect for the jth gene, $\tau_{jG_p(i)}$ represents a random group effect for observations from the $G_p(i)$th group (under pattern p) in the jth gene, and ε_{ji} represents a random error. Under expression pattern p, the density for the vector of log-scale observations for the jth gene, $f_p(\mathbf{y}_j|\boldsymbol{\theta})$, is evaluated according to a multivariate normal distribution with mean vector $(\mu, \ldots, \mu)'$ and pattern specific covariance matrix $\Sigma_p = \sigma^2 I + \sigma_\gamma^2 J + \sigma_\tau^2 M_p$ where I is the identity matrix, J is a matrix of 1's and

$M_p[i,j] = \begin{cases} 1 & \text{if } G_p(i) = G_p(j), \\ 0 & \text{otherwise.} \end{cases}$ This model has hyperparameters $\boldsymbol{\theta} = (\mu, \sigma^2, \sigma_\tau^2, \sigma_\gamma^2)$ and is a generalization of the LNN model. That is, the LNN model is a special case of the LN^3 model in which $\sigma_\gamma^2 = 0$.

2.2.3 The Lognormal-Normal model with gene-specific error variances

The LNN model assumes that all genes have a common error variance, σ^2. This assumption can be relaxed to allow each gene to have a unique error variance, σ_j^2, forming the LNNMV model. We consider three methods based on this model, including EBarrays' LNNMV.

Under this model, the log scale observation for the jth gene from the ith experimental unit under expression pattern p can be written as

$$y_{ji} = \mu + \tau_{jG_p(i)} + \varepsilon_{ji} \text{ where } \tau_{j1}, \ldots, \tau_{jn_p} \overset{iid}{\sim} N(0, \sigma_\tau^2)$$
$$\text{and } \varepsilon_{j1}, \ldots, \varepsilon_{jI} \overset{iid}{\sim} N(0, \sigma_j^2)$$

This model has hyperparameters $\boldsymbol{\theta} = (\mu, \sigma_\tau^2, \boldsymbol{\sigma}^2)$, where $\boldsymbol{\sigma}^2 = (\sigma_1^2, \sigma_2^2, \ldots, \sigma_J^2)$.

The LNNMV method from EBarrays places a scaled inverse chi-squared distribution on the gene-specific error variances. That is, $\sigma_j^2 \sim$ inv- χ^2 (df=ν, scaling=Φ), such that $\nu\Phi/\sigma_j^2 \sim \chi_\nu^2$. Given estimates $\hat{\nu}$ and $\hat{\Phi}$, the gene-specific error variances are estimated by $\hat{\sigma}_j^2 = \frac{\hat{\nu}\hat{\Phi} + (I-T)S_j^2}{\hat{\nu} + I - 2}$, where S_j^2 is the ordinary sample variance estimator with $(I-T)$ degrees of freedom for the log-scale observations from the jth gene and T is total number of experimental conditions.

The denominator of the LNNMV point estimator for σ_j^2 does not account for degrees of freedom used when estimating treatment means for each gene in the computation of S_j^2. Similar to MLEs for σ^2 in a traditional ANOVA analysis, this estimator systematically underestimates σ_j^2 resulting in liberal detection of differential expression. If one were to use a point estimator for σ_j^2, we would recommend the less liberal approach of using the posterior expectation $\hat{\sigma}_j^2 = \hat{E}(\sigma_j^2 | S_j^2) = \frac{\hat{\nu}\hat{\Phi} + (I-T)S_j^2}{\hat{\nu} + (I-T) - 2}$. We denote this approach as by LNNMV*; however, this adjusted denominator does not provide a fully adequate solution.

The EBarrays methods estimate the posterior probability that gene j follows expression pattern p given $\mathbf{y}_j$ as $\frac{\hat{\pi}_p f_p(\mathbf{y}_j | \hat{\boldsymbol{\theta}})}{\sum_{p=1}^{P} \hat{\pi}_p f_p(\mathbf{y}_j | \hat{\boldsymbol{\theta}})}$, assuming all hyperparameter estimates are the true hyperparameter values.

This expression is clearly sensitive to $\hat{\theta}$. Given that μ and σ_τ^2 are assumed to be the same for all genes and there are typically thousands of genes in a microarray data set, the effective sample size for estimating these parameters is high so that there will generally be little uncertainty associated with the ML estimates $\hat{\mu}$ and $\hat{\sigma}_\tau^2$ obtained from the EM algorithm. Therefore, it may be reasonable to act as if $\hat{\mu} = \mu$ and $\hat{\sigma}_\tau^2 = \sigma_\tau^2$ when estimating posterior pattern probabilities. Similarly, it may also be reasonable to ignore uncertainty in the estimator of σ^2 under the LNN and LN^3 models. However, when σ_j^2 is allowed to vary from gene to gene, there will be non-negligible uncertainty in the corresponding estimators $\hat{\sigma}_j^2$, which is not taken into account by assuming $\hat{\sigma}_j^2 = \sigma_j^2$. Under a model allowing for gene-specific error variances, a better estimator of the posterior probability that gene j follows expression pattern p is $\frac{\hat{\pi}_p f_p(\mathbf{y}_j|\hat{\mu},\hat{\sigma}_\tau^2,\hat{\nu},\hat{\Phi})}{\sum_{p=1}^{P}\hat{\pi}_p f_p(\mathbf{y}_j|\hat{\mu},\hat{\sigma}_\tau^2,\hat{\nu},\hat{\Phi})}$, where $f_p(\mathbf{y}_j|\hat{\mu},\hat{\sigma}_\tau^2,\hat{\nu},\hat{\Phi}) = \int f_p(\mathbf{y}_j|\hat{\mu},\hat{\sigma}_\tau^2,\sigma_j^2) f(\sigma_j^2|\hat{\nu},\hat{\Phi}) d\sigma_j^2$, where $f(\sigma_j^2|\hat{\nu},\hat{\Phi})$ is the empirically estimated inverse chi-squared prior distribution for σ_j^2.

Our suggested approach is to estimate $\hat{\nu}$ and $\hat{\Phi}$ using the method described by Smyth (2004) and compute shrunken estimates $\hat{\sigma}_j^2 = \hat{E}(\sigma_j^2|S_j^2)$ to use when fitting the EM algorithm to obtain estimates for μ, σ_τ^2, and π. Then when estimating the posterior expression pattern probabilities for each gene, we suggest empirically approximating $f_p(\mathbf{y}_j|\hat{\mu},\hat{\sigma}_\tau^2,\hat{\nu},\hat{\Phi})$ as $\sum_{q=1}^{Q} f_p(\mathbf{y}_j|\hat{\mu},\hat{\sigma}_\tau^2,\sigma*_q^2)/Q$ where $\sigma*_q^2$ is the $q/(Q+1)$th quantile of $f(\sigma_j^2|\hat{\nu},\hat{\Phi})$ and Q is a reasonably large number like 1000. We denote this method as LNNGV, which has hyperparameters $\theta = (\mu, \sigma_\tau^2, \nu, \Phi)$. The effectiveness and impact of this suggestion are examined in Sections 2.4-5 and Appendix B.

2.2.4 The Lognormal-Normal-Normal Model with gene-specific error variances

As with the LNN model, the LN^3 model assumes that all genes have a common error variance, σ^2, and this assumption can be relaxed to form the LN^{3}MV model, which allows for gene-specific error variances. For the LN^{3}MV model, we consider two methods, denoted LN^{3}MV* and LN^{3}GV, which incorporate gene-specific error variances in exactly the same way as the LNNMV* and LNNGV methods, respectively. The LN^{3}MV* (LN^{3}GV) method is a generalization of the LNNMV* (LNNGV) method. That is, the LNNMV* (LNNGV) method is a special case of the LN^{3}MV* (LN^{3}GV) method in which $\sigma_\gamma^2 = 0$.

2.3 Evidence Supporting Need for Three-level Hierarchical Models

Observations from a common gene are correlated for many reasons, even across differentially expressed treatments. Variability from gene to gene in several factors contribute to such correlation, including binding affinity of probe sets (Binder et al., 2004), amount of florescent dye that binds to each cDNA fragment (Binder et al., 2004), RNA transcription and degradation rate (Selinger et al., 2003), and the function of genes' corresponding proteins. These considerations imply that models for microarray data should contain gene effects like those present in the LN^3 and LN^3MV models but omitted from the models of Kendziorski et al. (2003).

The theoretical impact of gene effects when detecting DE genes can be demonstrated by comparing the modeled variance of differences between pairs of observations in two scenarios. The first scenario is when the observations in a pair come from different groups in a common gene. The second scenario is when the observations in a pair come from different genes. Under the LNN model, the variance of the difference for both scenarios is $2(\sigma_\tau^2 + \sigma^2)$. That is, the LNN model expects differences among same-gene observations from differentially expressed groups to "look like" differences among observations from different genes. However, when a gene effect is present, the variance for differences between observations from different genes is $2(\sigma_\gamma^2 + \sigma_\tau^2 + \sigma^2)$, which is greater than the variance for differences between observations from different groups in a common gene, $2(\sigma_\tau^2 + \sigma^2)$. In this case, the LNN model expects within-gene differences due to differential expression to be more extreme than they actually are, which reduces the model's power to detect differential expression. Creating a three-level hierarchical model by adding normally distributed gene effects is a tractable and effective method to correct this shortcoming. A similar argument can be made when considering models that accommodate gene-specific error variances.

If information about DE groups for each gene were known for real microarray data, we could check for evidence of gene effects by comparing the variance of between-gene differences to the variance of within-gene differences across DE groups. Because information about DE groups is unknown, such a simple strategy is not possible. However, we can fit three-level models to actual microarray data and examine the resulting estimates of σ_γ^2. Because the two-level models are special cases of three-level models with $\sigma_\gamma^2 = 0$, estimates of σ_γ^2 far from 0 provide evidence in favor of our proposed three-level

hierarchy over the two-level hierarchy. The next section presents results of two example microarray experiments where the estimates of σ_γ^2 provide clear support for the three-level hierarchy. We describe this point in detail in Appendix A.

As additional evidence of the inadequacy of models that omit gene effects, we compare the correlation structure implied by the LNN model to the correlation structure present in actual microarray data.

Under the LNN model, $cov(y_{ji}, y_{ji'}) = \begin{cases} \sigma_\tau^2 & \text{if } y_{ji} \text{ and } y_{ji'} \text{ are EE,} \\ 0 & \text{otherwise} \end{cases}$. For any two experimental units, under the LNN model, $\sum_{j=1}^{J} cov(y_{ji}, y_{ji'})/J = \pi_{EE}(i,i')\sigma_\tau^2$, where $\pi_{EE}(i,i')$ is the proportion of genes that are EE between experimental units i and i'. If experimental units i and i' correspond to the same experimental condition, an unbiased estimator of σ_τ^2 is given by $\hat{\sigma}_\tau^2(i,i') = \sum_{j=1}^{J}(y_{ji} - \bar{y}_{\cdot i})(y_{ji'} - \bar{y}_{\cdot i'})/(J-1)$, because $\pi_{EE}(i,i') = 1$ in this case. It follows that $\bar{\sigma}_\tau^2$ is also an unbiased estimator of σ_τ^2, where $\bar{\sigma}_\tau^2$ is the average of $\hat{\sigma}_\tau^2(i,i')$ over all pairs of experimental units (i,i') such that the experimental condition associated with experimental units i and i' is the same.

In practice, given an estimate $\hat{\pi}_{EE}(i,i')$, observed covariances between experimental units associated with different experimental conditions are often much larger than $\hat{\pi}_{EE}(i,i')\bar{\sigma}_\tau^2$. Table 2.2 summarizes this phenomenon for various treatment comparisons within two separate microarray data sets, which are described in Section 2.4. Each data set was analyzed with the LIMMA package for R developed by Smyth (2004). Estimates of $\pi_{EE}(i,i')$ were obtained by applying the method of Nettleton et al. (2006) to the distribution of p-values for each pairwise comparison. The final column provides estimates of between-treatment covariances, which were computed as the average of all the pairwise covariances involving one experimental unit from each of the two treatments. The LNN and LNNMV models imply the observed between-treatment covariances should closely match $\hat{\pi}_{EE}\bar{\sigma}_\tau^2$, but Table 2.2 shows that the estimated between-treatment covariances were larger than $\hat{\pi}_{EE}\bar{\sigma}_\tau^2$ for every treatment comparison.

The additional covariance observed between experimental units from different experimental conditions is easily explained by the presence of gene effects. For any two experimental units, under the LN^3 model, $\sum_{j=1}^{J} cov(y_{ji}, y_{ji'})/J = \sigma_\gamma^2 + \pi_{EE}(i,i')\sigma_\tau^2$ rather than $\pi_{EE}(i,i')\sigma_\tau^2$.

Table 2.2 Empirical evidence for presence of gene effects.

data set(Conditions)	$\hat{\pi}_{EE}$	$\bar{\sigma}^2_\tau$	$\hat{\pi}_{EE}\bar{\sigma}^2_\tau$	Average across Condition Cov
DC3000(NaCl,ctrl)	0.716	0.952	0.681	0.903
DC3000(phen,ctrl)	0.693	0.977	0.677	0.910
DC3000(PEG,ctrl)	0.352	0.914	0.322	0.838
DC3000(H_2O_2,ctrl)	0.961	0.957	0.920	0.948
Mouse(Ch,FF)	0.874	0.281	0.245	0.280
Mouse(Ch,MP)	0.824	0.281	0.231	0.279
Mouse(FF,MP)	0.956	0.284	0.272	0.284

2.4 Data Analysis

2.4.1 Data Set Descriptions

We analyzed a NimbleGen mRNA data set of 5608 genes from the DC3000 strain of the bacterial plant pathogen *Pseudomonas syringae* resulting from an unpublished experiment conducted in the Department of Plant Pathology at Iowa State University. NimbleGen performed RMA normalization on the data (Irizarry et al. (2003)). The experiment had two biological replicate samples grown in each of five different media: control (ctrl), phenol (phen), sodium chloride (NaCl), polyethylene glycol MW8000 (PEG), and hydrogen peroxide (H_2O_2). Before analyzing the data, the primary investigator suggested that any two noncontrol media will be EE only when they are also EE with the control, which reduces the number of expression patterns included in the analysis. Because each of the four treatments can be either EE or DE with the control, there are $2^4 = 16$ different expression patterns to consider.

The second data set we analyzed is a subset of the data used in Somel et al. (2008), available at the Gene Expression Omnibus (GEO) website as GDS3221. This experiment examined the impact of diet on the expression of 45101 genes in mice. We analyzed data from nine Affymetrix GeneChips corresponding to three treatment groups of three mice each. Each treatment involved ad libidum feeding of one of the following diets: (1) vegetables, fruit and yogurt identical to the diet fed to chimpanzees in their ape facility (Ch); (2) McDonald's fast food (FF); (3) mouse pellets on which the mice were raised (MP). To keep the presentation simple, we have omitted data from a second batch of chips and a fourth diet group (cafeteria food), which produced expression profiles very similar to those from the McDonald's diet. With the three included treatment groups, there are a total of five possible expression

patterns: Ch=FF=MP; Ch=FF≠MP; Ch≠FF=MP; Ch=MP≠FF; and Ch≠FF, Ch≠MP, FF≠MP.

2.4.2 Analysis of Real Data

We analyzed these data sets with each of the eight methods and report the resulting parameter estimates from the GG, LNN, LNNGV, LN^3 and LN^3GV methods in Table 2.3. (The LNNMV*, LNNMV, and LN^3MV* methods share theoretical models (and thus parameter estimates) with the LNNGV, LNNGV, and LN^3GV methods, respectively.) The parameter estimates in Table 2.3 are consistent with what we expected. For both data sets, when a random gene effect is accounted for in the model, the estimated treatment effect variance decreases drastically and the gene effect variance is estimated to be much larger than the treatment effect variance. This means the LN^3 and LN^3GV methods are able to detect smaller treatment effects than their respective two-level counterparts, LNN and LNNGV. It is not surprising then to see that for both data sets the LNN method estimates a larger proportion of genes following the null pattern than does the LN^3 method, or that the LNNGV method estimates a larger proportion of genes following the null pattern than does the LN^3GV method.

Rather than examining parameter estimates, researchers are often more interested in creating lists of genes that are likely to follow expression patterns of interest. To construct a list of DE genes, one would collect all genes with an estimated posterior probability of equivalent expression (ePPEE) less than a given threshold. When the ePPEE falls below the given threshold for many genes, not all identified potentially DE genes may be individually studied further. However, the size and contents of the entire list provides important information to researchers about the global effects of the treatments on gene expression. The composition of a long list of potentially DE genes forms the basis for popular gene set enrichment analyses that are commonly used to interpret the results of microarray experiments. To examine the practical differences between gene lists created by the methods, we begin by plotting the empirical CDF of the ePPEEs for each method for the two data sets in Figure 2.4.2. These plots quickly provide the observed size of a gene list for any PPEE cutoff, obtained by intersecting a vertical line at the desired PPEE cutoff with the curve for each method.

The plots show substantial differences between the examined methods in the number of detected genes over a wide range of PPEE thresholds. For models with gene-specific error variances, incorporating uncertainty in estimated error variances greatly reduced the number of detected genes (LNNGV and

Table 2.3 Hyperparameter estimates and estimated proportion of null genes for DC3000 (top) and mouse diet (bottom) data from each of the models.

	Model Used to Analyze				
Parameter	GG	LNN	LNNGV	LN^3	LN^3GV
$\hat{\alpha}$	69.8	-	-	-	-
$\hat{\alpha}_0$	1.54	-	-	-	-
$\hat{\nu}$*	0.0254	-	-	-	-
$\hat{\mu}$	-	0.501	0.419	0.277	0.264
$\hat{\sigma}^2_\tau$	-	0.982	0.878	0.151	0.101
$\hat{\sigma}^2_\gamma$	-	-	-	0.813	0.832
$\hat{\sigma}^2$	-	0.0129	-	0.0116	-
$\hat{\Phi}$	-	-	0.00509	-	0.00509
$\hat{\nu}$	-	-	3.546	-	3.546
$\hat{\pi}_{null}$	0.728	0.721	0.657	0.655	0.492
$\hat{\alpha}$	269.5	-	-	-	-
$\hat{\alpha}_0$	4.59	-	-	-	-
$\hat{\nu}$*	0.0187	-	-	-	-
$\hat{\mu}$	-	0.206	0.210	0.194	0.194
$\hat{\sigma}^2_\tau$	-	0.279	0.281	0.00468	0.00678
$\hat{\sigma}^2_\gamma$	-	-	-	0.278	0.275
$\hat{\sigma}^2$	-	0.00346	-	0.00331	-
$\hat{\Phi}$	-	-	0.00249	-	0.00249
$\hat{\nu}$	-	-	8.186	-	8.186
$\hat{\pi}_{null}$	0.958	0.954	0.931	0.802	0.840

LN^3GV curves are lower than LNNMV* and LN^3MV* curves, respectively). In the DC3000 data at a PPEE cutoff of 0.1, for example, the LNNMV, LNNMV* and LNNGV methods would produce lists with 1983, 1498, and 893 genes, respectively. Incorporating gene effects greatly increased the number of detected genes (LN^3, LN^3MV*, and LN^3GV curves are higher than LNN, LNNMV*, and LNNGV curves, respectively). In the mouse diet data at a PPEE cutoff of 0.1, for example, the LN^3GV method identified almost three times as many DE genes as the LNNGV method (945 vs. 324 genes, respectively). These results indicate that differences between the methods' ePPEEs are practically significant, and care should be taken when choosing among the suggested methods.

Constraints on time, money, material, and personnel resources limit the number of genes that researchers will follow up on with further study. Thus, the overlap between lists from each method containing a fixed number of the most significant genes is an important feature for assessing the simi-

larity between methods' results. Table 2.4 provides the size of pairwise intersections of lists containing the 200 most significant genes from each method for the DC3000 and mouse diet data sets, respectively. These results show substantial practical differences between rankings, as many lists overlap by roughly half their genes.

Table 2.4 Overlap in lists of top 200 most significant DE genes for DC3000 (top) and mouse diet (bottom) data.

Method	1	2	3	4	5	6	7
(1)GG	200						
(2)LNN	187	200					
(3)LNNMV	122	119	200				
(4)LNNMV*	118	120	160	200			
(5)LNNGV	130	127	185	162	200		
(6)LN^3	186	198	117	118	125	200	
(7)LN^{3}MV*	117	114	194	154	184	113	200
(8)LN^{3}GV	77	81	137	149	133	79	135

Method	1	2	3	4	5	6	7
(1)GG	200						
(2)LNN	193	200					
(3)LNNMV	108	107	200				
(4)LNNMV*	125	124	152	200			
(5)LNNGV	88	87	173	136	200		
(6)LN^3	193	197	109	124	89	200	
(7)LN^{3}MV*	93	92	181	134	184	94	200
(8)LN^{3}GV	83	82	155	148	158	82	148

2.5 Simulation Study

Here we briefly summarize our simulation study and its results. Detailed accounts of simulation procedures and results are presented in Appendix B.

We conduct a variety of simulations to assess the accuracy and power of the considered methods. By "accuracy," we refer to the property that for any given collection of genes the average estimated posterior probability for each pattern should closely match the proportion of genes in the collection that follow the given pattern. By "power," we refer to a method's ability to detect differential expression.

We prefer the method that creates the largest list of genes for a given ePPEE threshold, provided that the method's ePPEEs are accurate.

We simulated data from each of the five models (GG, LNN, LNNMV, LN^3 and LN^3MV) using the model parameters reported for the DC3000 data set in Table 2.3. In addition to these model based simulations, we also conducted simulations using an HIV mRNA expression data set from the GEO website, named GDS1449. We analyzed each simulated data set with each method and recorded estimated posterior probabilities for each expression pattern for each gene.

The simulation results clearly support our claims that failing to distinctly model gene and gene-specific treatment effects reduces power and produces conservative results and that using point estimates of error variances produces liberal results. The LN^3GV method stands out as the best method from these simulations. The LN^3GV method was the only method to produce accurate ePPEEs in all simulation scenarios, and no method produced better average significance rankings (as seen in ROC curves) than those of the LN^3GV method in any simulation scenario. The LN^3GV method exhibited greater power than the LNNGV method, which was the only other method that did not produce liberal results in at least one simulation scenario.

2.6 Discussion

When modeling a data set that includes multiple observations from each of multiple genes, a conventional analysis would begin with a model that incorporates gene effects. One might decide to omit gene effects if, after looking, there was no evidence of gene effects or if results from an analysis were not affected by the omission of gene effects. We have demonstrated that gene effects are present in real data sets and provided generalizations of the methods based on lognormal two-level hierarchical models to include gene effects. These generalizations behave nearly identically to their two-level counterparts when analyzing data without gene effects and improve power and accuracy when data contain gene effects. These extensions serve as an example of how other hierarchical models that omit gene effects might be improved by more versatile modeling.

Using point estimates of gene-specific error variances without accounting for their uncertainty produces liberal ePPEEs. We have suggested a corrected approach that involves integration over an em-

pirically estimated prior distribution for the error variances and demonstrated this adaptation yields accurate ePPEEs.

As noted in the introduction, we have identified nearly a dozen methods that omit gene effects. There are far more methods in the microarray data analysis literature that do not suffer from this problem. Most published methods explicitly or implicitly include gene effects whose distribution is allowed to differ from the distribution of within gene treatment effects. Methods based on gene-specific linear models that make no attempt to borrow information across genes fall into this category, as do methods that borrow information across genes only for the purpose of improved error variance estimation. While we expect our LN^{3}GV method to perform well when compared against the large collection of competing approaches, a broad comparison of methods is beyond the scope of this paper, and we make no claims of superiority here. Our main point is that the hierarchical modeling approach pioneered by Kendziorski et al. (2003) can be improved by the inclusion of both gene and gene-specific treatment effects. Given the influential nature of the original work of Kendziorski et al. (2003), we think this is an important point to make.

The development of the LNN and GG models by Kendziorski et al. (2003) represents groundbreaking work on the hierarchical modeling of gene expression data. We have shown how to improve on the original work by allowing for random gene effects and replacing gene-specific error variance point estimates without dramatically affecting computational costs. Adding random gene effects to a model increases the dimension of the parameter space across which the EM algorithm must optimize by one, but does not substantially increase computational costs. For any of the described methods, analyzing a data set with 5000 genes, 9 experimental units and 4 expression patterns of interest takes less than 10 minutes using a single Linux machine and R code that calls a C routine to evaluate multivariate normal densities. We have developed the R package LN3GV (available at the CRAN webpage) to implement the LNNMV*, LNNGV, LN^3, LN^{3}MV* and LN^{3}GV methods discussed in this article. Throughout this paper, the GG, LNN, and LNNMV methods were implemented via the EBarrays package.

We have focused on the approach of Kendziorski et al. (2003) not only because of its influential nature but also because of its unique and elegant approach to inference for experiments with more than two treatments. The vast majority of competing approaches have been developed primarily for the case of two treatments. While it is easy to extend many of these methods to cover the case of more than two

treatments, very few methods outside the Kendziorski et al. (2003) lineage provide an inherent natural strategy for classifying genes according to their pattern of expression across multiple treatments. Thus, we believe our efforts to improve the original work of Kendziorski et al. (2003) have been well spent.

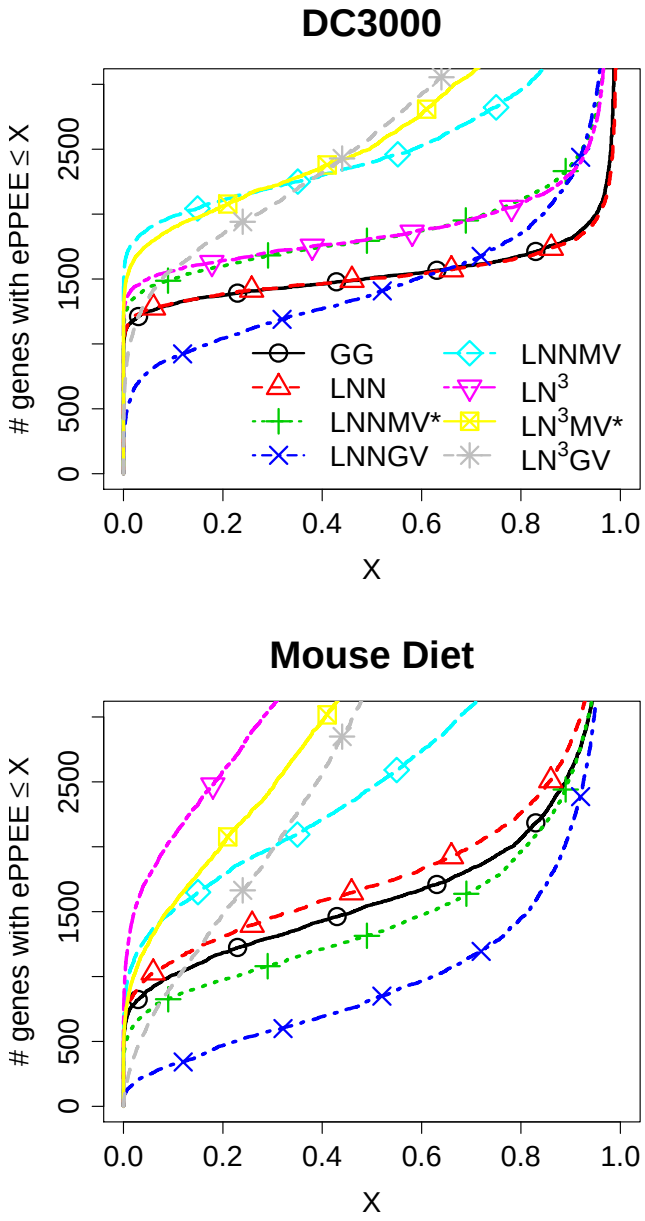

Figure 2.1 Comparison across methods of empirical ePPEE CDFs for DC3000 (top) and mouse diet (bottom) data.

A. Evidence Supporting Need for Three-level Hierarchical Models

The LN^{3}MV model parameter estimates for the DC3000 and mouse diet data sets provide further evidence of the presence of gene effects in real data. To demonstrate, we analyzed a wide variety of data simulated from two-level hierarchical models and examined values of $\hat{\sigma}^2_\gamma$ obtained from analysis with the LN^{3}MV method. We simulated from the LNN and LNNMV models for several different values of σ^2_τ. We simulated from the GG model for several values of α_0 and, separately, for several values of $\nu*$. All other model parameters were fixed at the corresponding value estimated from the DC3000 data set analysis. The experimental design and expression pattern structure used for these simulations was the same as for the simulations described in Section 2.5.

We also simulated from the HIV data set under the structure of the LNNMV model. That is, $y_{ji} = \mu + \tau_{jG_{p(j)}(i)} + \varepsilon_{ji}$ where $p(j)$ is the desired expression pattern for the jth gene, μ represents the average expression of all genes and groups, $\tau_{jG_{p(j)}(i)}$ represents a random group effect for observations from the $G_{p(j)}(i)$th group in the jth gene, and ε_{ji} represents a random error. To generate the data for simulation b, we randomly ordered the genes and randomly picked 9 subjects from the pool of 23 subjects who were identical with regard to the factors considered in the HIV study. Let $\mathbf{y}^{(b)}$ be the resulting 8793x9 matrix of log-scale observations for simulation b. We constructed error terms by subtracting row averages of this matrix from their corresponding observations ($\varepsilon^{(b)}_{ji} = y^{(b)}_{ji} - \bar{y}^{(b)}_{j\cdot}$). We randomly constructed each group effect using the (scaled) difference between the average of a randomly sampled row and the overall average. That is, $\tau^{(b)}_{jG_{p(j)}(i)} = c^{(b)}(\bar{y}^{(b)}_{j'\cdot} - \bar{y}^{(b)}_{\cdot\cdot})$ where j' is a randomly selected integer between 1 and 8793 and $c^{(b)}$ is a constant chosen to determine the scale of the group effects' distribution in simulation b. Finally, we constructed simulated data as $y^{sim(b)}_{ji} = \bar{y}^{(b)}_{\cdot\cdot} + \tau^{(b)}_{jG_{p(j)}(i)} + \varepsilon^{(b)}_{ji}$ and used the first 5001 rows of the resulting matrix. This approach ensured that differences due to treatment effects are distributed identically to differences across genes, as in each of the two-level hierarchical models.

The simulation results presented in Figure A.1 show the LN^{3}MV model produced smaller estimates

of σ_γ^2 for the data simulated without gene effects than for the actual data sets. There was very little variability among the estimates of σ_γ^2 for simulated data sets where $\hat{\sigma}_\tau^2$ is less than 0.5. Also, $\hat{\sigma}_\tau^2$ was substantially less than $\hat{\sigma}_\gamma^2$ for every simulated data set, but the opposite was true for both of the actual data sets. This evidence supports our belief that there are often substantial gene effects present in real microarray data, which are not appropriately accounted for by any of the two-level hierarchical models.

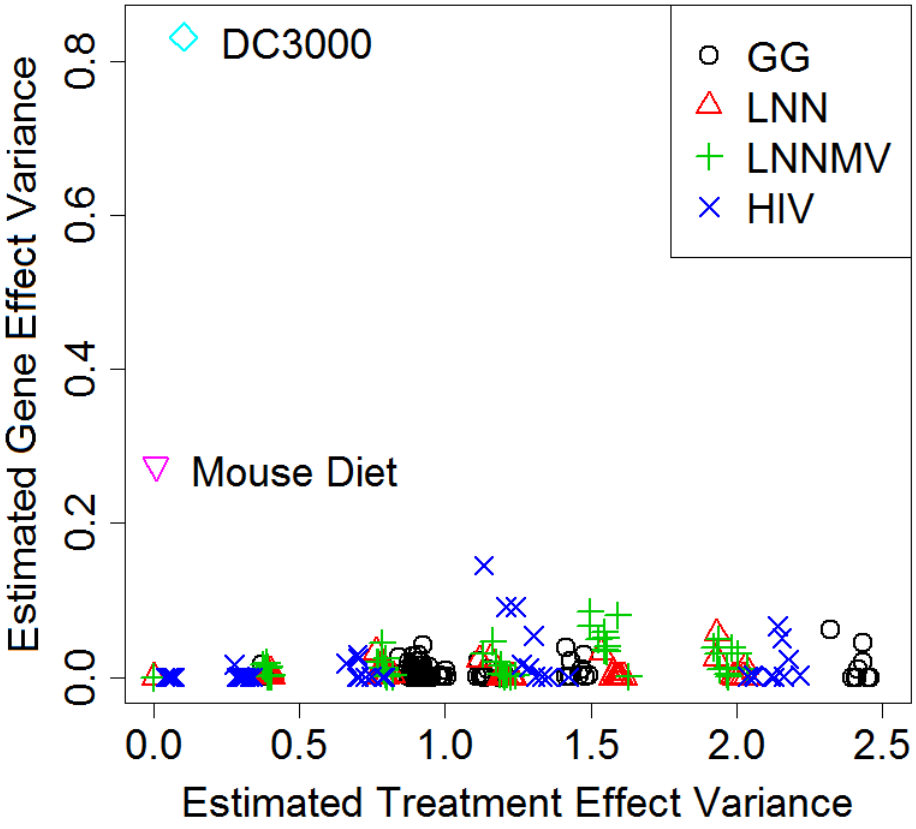

Figure A.1 LN^{3}GV model estimates of σ_γ^2 and σ_τ^2 for simulated and real microarray data.

B. Simulation Study

We conducted a variety of simulations to assess the accuracy and power of the considered methods. By "accuracy," we refer to the property that for any given collection of genes the average posterior probability for each pattern should closely match the proportion of genes in the collection that follow the given pattern. By "power," we refer to a method's ability to detect differential expression. We prefer the method that creates the largest list of genes for a given PPEE threshold, provided that the method's ePPEEs are accurate.

To examine the effectiveness of the proposed methods, we conducted a series of simulations for an experimental setup with 3 experimental conditions (1 control and 2 treatments) each with 3 replicates. As with the DC3000 data analysis in the main text, our expression patterns of interest compared each treatment with the control. Thus, the simulations have $2^2 = 4$ different expression patterns. Each simulation scenario was repeated 10 times.

B.1 Model Based and Data Based Simulation Studies

We simulated data from each of the five models (GG, LNN, LNNMV, LN^3 and LN^3MV) using the model parameters reported for the DC3000 data set in Table 2.3. In addition to these model based simulations, we also conducted simulations using an HIV mRNA expression data set from the GEO website, named GDS1449. Simulations based on real data examine model performance in more realistic scenarios where data include dependence across genes and do not perfectly follow known distributions. The HIV data set contained expression data for 8793 genes from each of 87 subjects, 23 of whom shared a common treatment group. For each simulation based on the HIV data set, we randomly sampled 9 subjects from the subset of 23 and sampled 5001 genes from the 8793 total genes in the data set. The treatment effect for the mth group in the jth gene was simulated independently and identically

as $\tau_{mj} = t_{mj}2(b_{mj} - .5)$ where $t_{mj} \overset{iid}{\sim}$ gamma(shape $\alpha = 1$, rate $\beta = 1.25$) and $b_{mj} \overset{iid}{\sim}$ Bernoulli(.5). The treatment effects were added to the HIV data to create the desired expression patterns. When analyzed by the LN^{3}GV method, the 20 data sets produced by this simulation method had parameter estimate averages (and standard errors) $\hat{\mu} = 5.33(0.005), \hat{\sigma}_\gamma^2 = 1.539(0.022), \hat{\sigma}_\tau^2 = 0.960(0.016), \hat{\nu} = 3.284(0.027), \hat{\Phi} = 0.255(0.004)$.

For every simulation, 3000 genes were simulated to follow the null pattern (all treatments are EE with the control) and 667 genes for each of the other expression patterns of interest. We then analyzed each simulated data set with each method and recorded estimated posterior probabilities for each expression pattern for each gene.

B.2 Simulation Results

The LNN, LNNMV*, and LNNGV methods are special cases of the LN^3, LN^{3}MV*, and LN^{3}GV models in which $\sigma_\gamma^2 = 0$. For data simulated from models without gene effects, we would expect estimated expression pattern posterior probabilities from the LNN, LNNMV*, and LNNGV methods to closely match those from the LN^3, LN^{3}MV*, and LN^{3}GV models, respectively. All correlations between the PPEEs estimated from the LNN, LNNMV*, and LNNGV methods and the PPEEs estimated from the LN^3, LN^{3}MV*, and LN^{3}GV methods, respectively, were greater than 0.998 for each of the 10 simulations from each of the GG, LNN, and LNNMV models. This demonstrates that the methods based on three-level hierarchical models can adapt to data without gene effects and perform nearly identically to two-level methods when data are generated from a two-level hierarchical model. Thus, our results indicate that there is no harm in using a three-level method, even if the true data generating mechanism is consistent with a two-level hierarchy. In contrast, many of the results that we shall present subsequently show that two-level methods can perform quite poorly relative to three-level methods when the true data generating mechanism is consistent with a three-level hierarchy.

For any given collection of genes, the average estimated posterior probability for each pattern should closely match the proportion of genes in the collection that follow the given pattern. We focus in particular on collections of genes that have been assigned small estimated posterior probabilities of equivalent expression (ePPEEs) as these are of primary interest in practice. To examine the methods

with respect to this criterion, for each simulated data set, we sorted genes according to their ePPEEs, from smallest to largest. Beginning with the 100 genes with the smallest ePPEEs, we created lists by adding genes with the smallest ePPEE one at a time and plotting the observed proportion of listed genes that were actually simulated from the null expression pattern versus the average ePPEE for the listed genes. An ideal method should produce curves that closely follow the $y = x$ diagonal. Curves appearing substantially above (below) the $y = x$ diagonal in the plotted range are considered liberal (conservative) relative to their reported posterior probability estimates.

Figure B.1 displays results from applying each method to ten separate simulations from the HIV data set and each of the five models. The plots for the LNNMV simulation indicate that methods using point estimates of gene-specific error variances (LNNMV, LNNMV*, and LN^{3}MV*) produced liberal ePPEEs, as did methods assuming a constant coefficient of variation (GG, LNN, and LN^3). EBarrays' LNNMV method was most liberal because it uses a biased gene-specific variance estimator that tends to underestimate each true variance. The LNNMV, LNNMV*, and LN^{3}MV* methods underestimated the proportion of null genes appearing on a gene list for several simulation models, including LNNMV. That is, EBarrays' LNNMV method produced liberal posterior probability estimates even when analyzing data simulated from its own model. Liberal posterior probability estimates lead to misplaced confidence in the underlying expression pattern for a given gene, which is problematic when creating gene lists with controlled error rates. The LNNGV and LN^{3}GV curves closely follow the $y = x$ diagonal for the LNNMV simulations, which demonstrates the validity of our suggested method for handling gene-specific error variances.

The plots for the LN^3 and LN^{3}MV simulations indicate that methods omitting gene effects produced conservative ePPEEs when gene effects are present in the data. When analyzing data simulated from the LN^{3}MV model, the LNNMV and LNNMV* methods suffer from two separate, counteracting issues: omitting gene effects, which leads to conservative ePPEEs, and using point estimates of gene-specific error variances, which leads to liberal ePPEEs. With the exception of the HIV data set simulations, the LN^{3}GV curves closely follow the $y = x$ diagonal for all simulation scenarios, indicating this method produced robustly accurate ePPEEs.

There is substantial variability between the curves in each plot for simulations from the HIV data set. It is, nonetheless, clear that the LNNGV and LN^{3}GV methods were the only methods that did not

produce blatantly liberal posterior probability estimates. The LNNGV appeared to be slightly conservative, while the LN^3GV was not clearly biased.

We next examine the power of each method. A method that says all expression patterns are equally likely for every gene is not very useful. Assuming the posterior probability estimates are accurate as discussed above, a method that strongly differentiates among the expression patterns for as many genes as possible is preferable. That is, we would prefer the method that creates the largest list of genes for a given ePPEE threshold, provided that the method's ePPEEs are accurate. To examine the methods with respect to this criterion, we sorted genes according to their ePPEEs from smallest to largest. We then plotted the average ePPEE for each rank across the ten simulation iterations versus rank. (Averaging by rank allows us to create one plot for all simulation iterations and to use standard error lines, rather than requiring a separate plot for each simulation iteration or model.) These plots quickly provide the estimated size of a gene list for any ePPEE cutoff, given by intersecting a vertical line at the desired cutoff with the curves for each method. Figure B.2 displays the average results for LNNGV, LN^3GV, and LNNMV methods applied to ten simulations from the HIV data set and the LN^3MV model. Dashed lines are $\pm$ two standard errors. There are substantial differences between the curves. While the LNNMV method tends to place more genes on a list for PPEE thresholds near zero, this method was demonstrated to give substantially liberal ePPEEs. In both cases, the LN^3GV method places noticeably more genes on a DE list for a given threshold than does the LNNGV method.

Finally, we examine the significance rankings of each method by creating receiver operating characteristic (ROC) curves. The solid curves in Figures B.3-5 display ROC curves averaged across ten simulations from each of the models. The dashed lines are $\pm$ two standard errors around the mean, providing approximate 95% confidence intervals. The ROC curves for the HIV, LN^3MV, and LNNMV simulations are widely separated into two groups with curves for methods assuming a constant coefficient of variation falling substantially beneath curves for methods allowing for gene-specific error variances. In the LN^3 simulations, the ROC curves for the methods based on two-level models fall slightly beneath curves for methods based on three-level models. The ROC curves for data simulated from the LNN and GG models show little difference between any of the methods, with the exception that the LNNMV method may produce a curve slightly lower than those of the other methods. No method produces ROC curves higher than those produced by the LN^3GV method for any considered

simulation. In general, these plots demonstrate that in addition to producing accurate, powerful ePPEEs, the LN^{3}GV method also produces improved ROC curves that are robust to model misspecification.

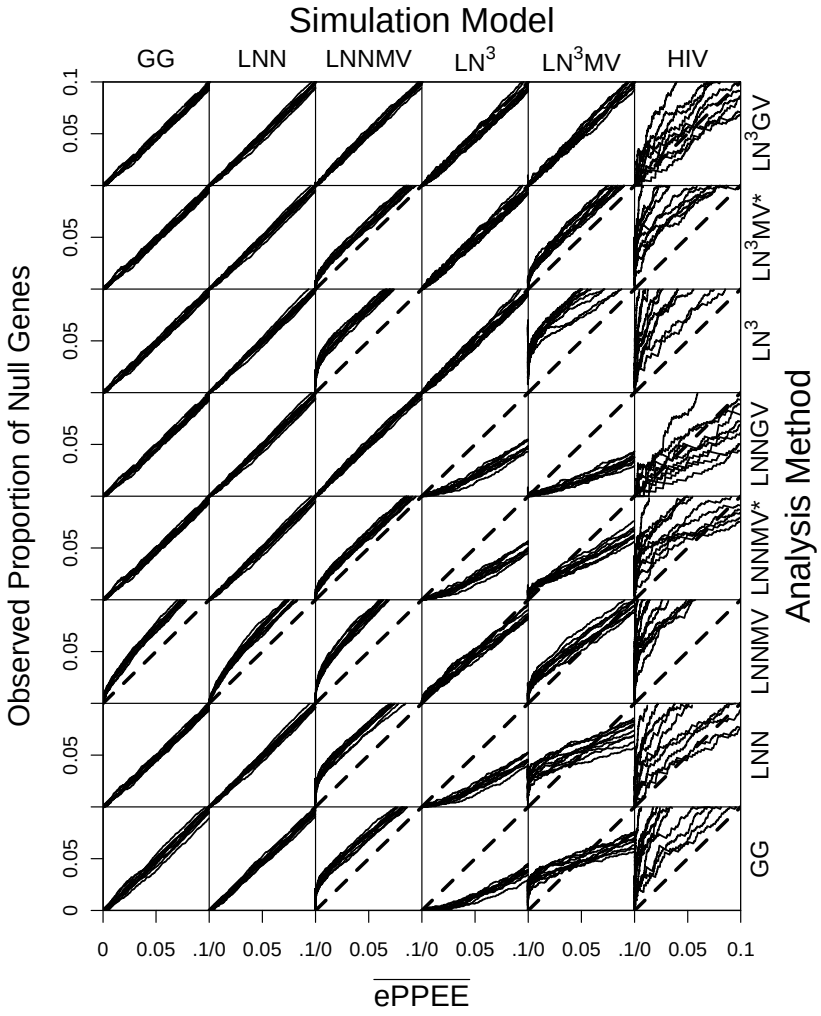

Figure B.1 Observed proportion of null genes vs. $\overline{\text{ePPEE}}$ for simulated data.

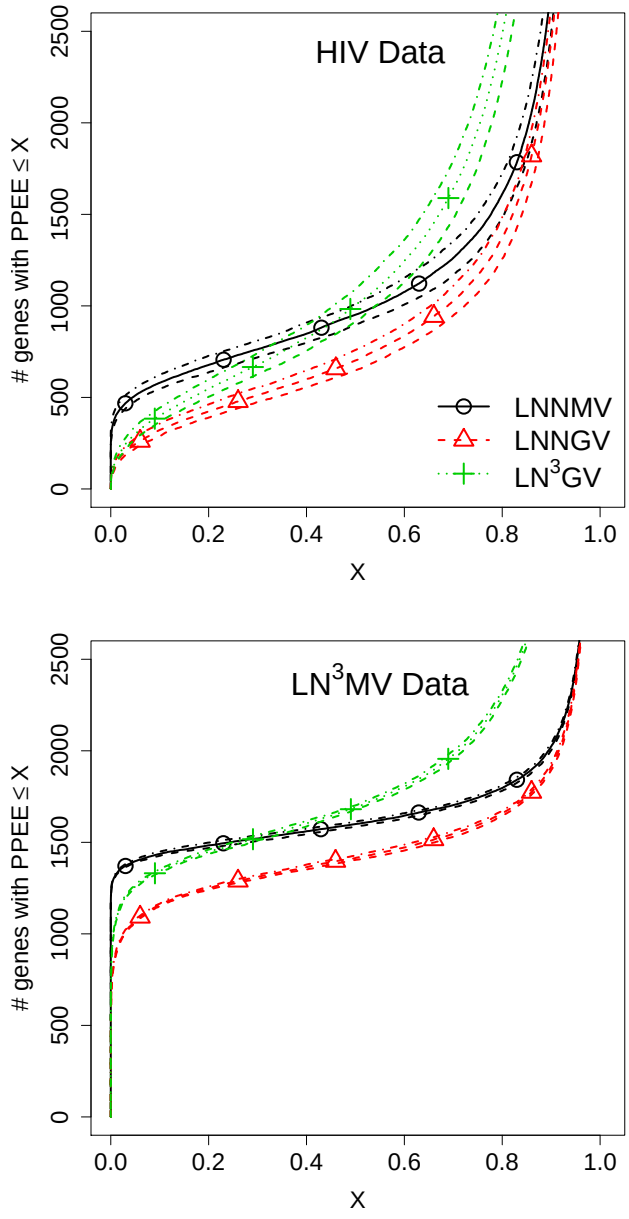

Figure B.2 Comparison across methods of empirical ePPEE CDFs for simulations from HIV data (top) and LN^3MV model (bottom).

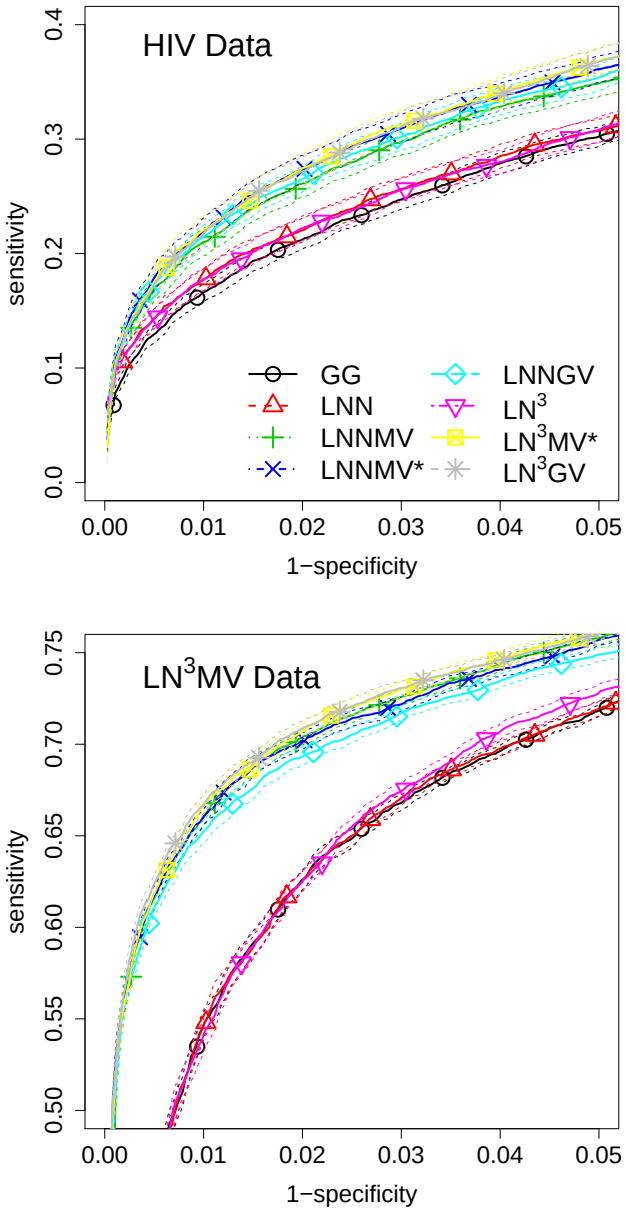

Figure B.3 ROC curves for simulations from HIV data (top) and LN^3MV model (bottom).

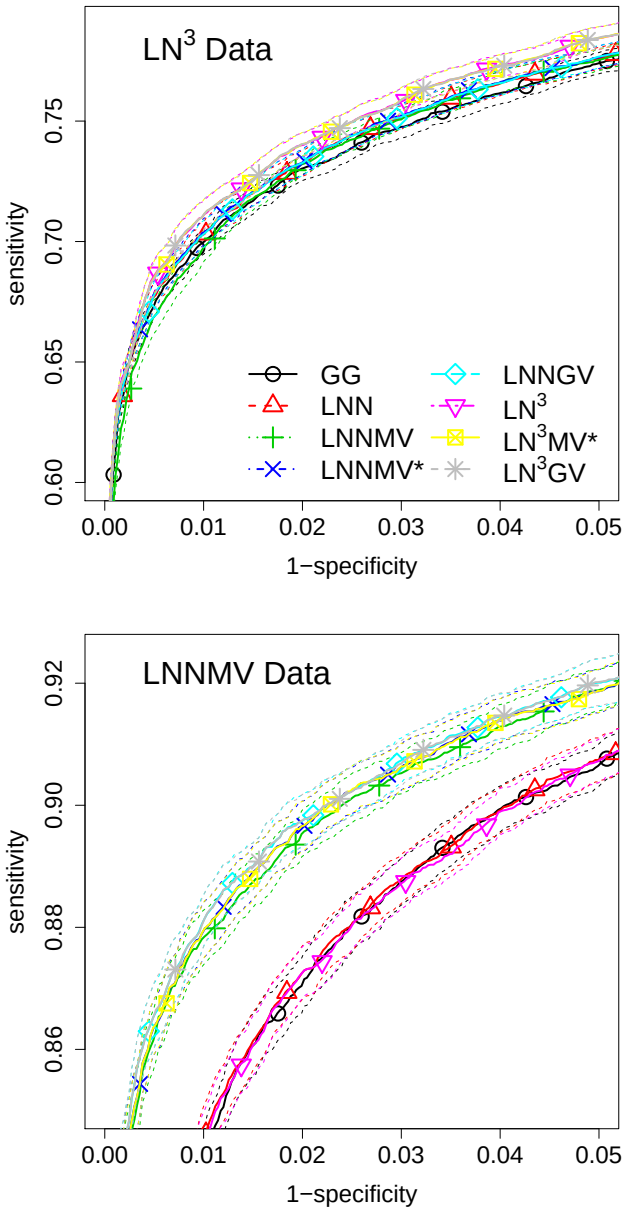

Figure B.4 ROC curves for simulations from LN^3 (top) and LNNMV (bottom) models.

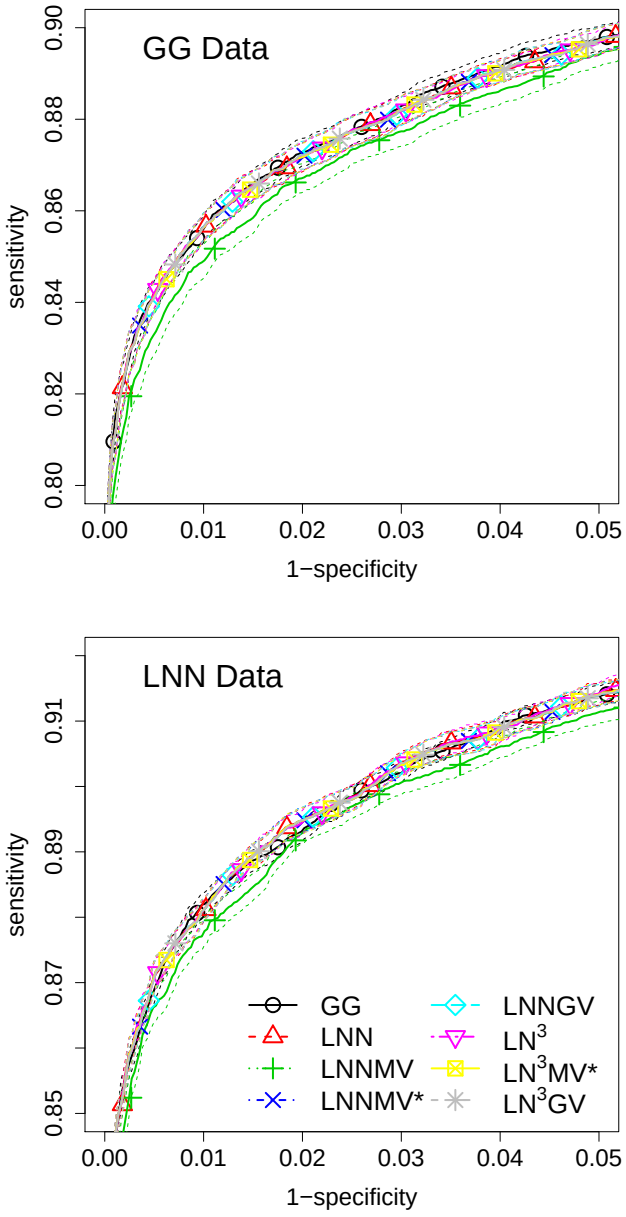

Figure B.5 ROC curves for simulations from GG (top) and LNN (bottom) models.

CHAPTER 3. DETECTING DIFFERENTIAL EXPRESSION IN RNA-SEQUENCE DATA USING QUASI-LIKELIHOOD WITH SHRUNKEN DISPERSION ESTIMATES

A paper submitted to *Statistical Applications in Genetics and Molecular Biology*

Steven P. Lund, Dan Nettleton, Davis McCarthy, and Gordon Smyth

Abstract

Next generation sequencing technology provides a powerful tool for measuring gene expression (mRNA) levels in the form of RNA-sequence data. Method development for identifying differentially expressed (DE) genes from RNA-seq data, which frequently includes many low-count integers and can exhibit severe overdispersion relative to Poisson or binomial distributions, is a popular area of ongoing research. Here we present quasi-likelihood methods with shrunken dispersion estimates based on an adaptation of Smyth's (2004) approach to estimating gene-specific variances in LIMMA for microarray data. Our suggested methods are computationally simple and analogous to ANOVA and compare favorably versus competing methods in detecting DE genes across a variety of simulations based on real data.

3.1 Introduction

Next-generation sequencing (NGS) technologies are powerful and increasingly popular tools used to identify differentially expressed genes, among other gene expression characteristics. RNA-Seq and SAGE technologies provide discrete count data serving as measures of messenger RNA (mRNA) expression levels through the following procedure. The mRNA is isolated from sample cells, fragmented, and copied to complementary DNA (cDNA). The cDNA fragments are then amplified and sequenced, and the resulting reads are aligned with a reference genome. The number of reads mapped within each

reference gene provides the RNA-Seq count data. This paper considers integer counts, typically ranging from zero to many thousands, of single-end reads that uniquely map to a single gene. Because of the frequent presence of low integers, methods developed for analyzing microarray data, which can be modeled as a continuous response, are not generally appropriate for analyzing RNA-Seq data.

As NGS has grown in popularity among researchers exploring differential expression, many statistical methods have been proposed for handling the subsequent expression data. Poisson or binomial (with n fixed as the sample library size) generalized linear models (GLM) could certainly handle low integer counts present in RNA-Seq data. However, upon modeling data with biological replicates within experimental conditions, it is clear that the restrictive mean-variance relationships for the Poisson and binomial distributions do not adequately accommodate the variability present in RNA-Seq data. That is, the RNA-Seq data are overdispersed, exhibiting greater variability across biological replicates than Poisson or binomial models predict.

In the face of overdispersion, one option is to add random effects to the original GLM, creating a generalized linear mixed effects model, as demonstrated by Blekhman et al. (2010). Another option is to choose a more flexible distribution. Zhou et al. (2011) and Vêncio et al. (2004) use beta-binomial models to account for overdispersion. Several methods, including Lu et al. (2005); Robinson and Smyth (2007, 2008); McCarthy et al. (2012); Anders and Huber (2010); Di et al. (2011), are based on the negative-binomial distribution, which has two parameters and a more flexible mean-variance relationship than the Poisson or binomial (with fixed n) distributions. Although the negative binomial distribution provides flexibility in modeling variances, existing popular methods based on this distribution fail to adequately account for uncertainty in parameter estimates. A simulation study described in Section 3.4 demonstrates that these methods produce an over-abundance of small p-values for tests with true null hypotheses, relative to a uniform distribution, even for data simulated from negative binomial distributions. These non-uniform distributions of p-values from tests with true nulls are shown to produce q-values (Storey and Tibshirani, 2003) that underestimate false discovery rates.

Tjur (1998) describes a general use quasi-likelihood (QL) approach to adjusting for overdispersion. To implement Tjur's method for RNA-Seq data, average counts for observations from the kth gene are modeled in the typical GLM fashion by specifying covariates and a link function. The variance of each observation from gene k is assumed to be a user-specified function of the modeled averages, multiplied

by a gene-specific dispersion parameter denoted by Φ_k. The QL approach then compares the ratio $LRT_k/(q\hat{\Phi}_k)$ to an appropriate F-distribution, where LRT_k is a quasi-likelihood ratio test statistic for the kth gene, q is the difference between the dimensions of the full and null-constrained parameter spaces, and $\hat{\Phi}_k$ is an estimate of the dispersion for the kth gene. Auer and Doerge (2011) suggest a two-stage Poisson model (TSPM), which first tests each gene for overdispersion (i.e. $\Phi_k > 1$) and then adjusts a Poisson model likelihood ratio test (LRT) for significantly overdispersed genes using a form of Tjur's QL method.

A drawback to using Tjur's QL approach with RNA-Seq data is that while many methods exist for estimating the dispersion for a single gene, there are often few degrees of freedom available for these estimates. In Section 3.2, we propose adapting Smyth's (2004) approach to estimating gene-specific error variances for microarray data in order to share information across genes when estimating gene-specific dispersion parameters for the QL approach. The resulting new methods are powerful, robust and fast, and accommodate all experimental designs that can be analyzed by an ordinary GLM. These suggested QL methods are analogous to ANOVA with shrunken variance estimates, where deviances are analogs to sums of squares.

In Section 3.3, we apply our new methods to real RNA-Seq data and compare results with several other popular methods. Section 3.4 describes simulation studies that demonstrate our recommended approach offers significantly improved differential expression significance rankings and better estimates of false discovery rates compared to competing methods when analyzing RNA-Seq datasets with small to moderate sample sizes common in practice. The authors provide brief commentary regarding the suggested methods and alternative methods based on exact tests in Section 3.5. Section 3.6 provides example code for analyzing two datasets with the suggested methods of this article via the R (R Development Core Team, 2011) package QuasiSeq.

3.2 Method Description

3.2.1 Review of Related Methods

Auer and Doerge (2011) developed a quasi-likelihood approach for analyzing RNA-Seq data called TSPM. This approach first tests each gene for overdispersion, relative to a fitted Poisson model, and then

adjusts the likelihood ratio test (LRT) for significantly overdispersed genes using a form of Tjur's QL method. Our simulation studies in Section 3.4 show that this approach will tend to correct for overdispersion only when it is severe and that this can lead to very liberal tests for differential expression. The proposed methods in this article use a more conservative approach to adjusting for overdispersion and provide the additional advantage of sharing information across genes when estimating dispersions.

The negative binomial distribution is popular among methods for analyzing RNA-Seq data. (See, for example, edgeR (Robinson and Smyth, 2007, 2008; Robinson et al., 2010; McCarthy et al., 2012), DESeq (Anders and Huber, 2010) and NBPSeq (Di et al., 2011), which all use negative binomial models to analyze RNA-Seq data.) For a detailed review of these methods, see McCarthy et al. (2012). While offering several ways to estimate negative binomial dispersion parameters, these methods all treat the resulting estimates as known constants when testing for differential expression and can be shown to produce liberal p-values, with the exception of DESeq for which the distribution of p-values is often J-shaped. Among the popular methods based on the negative binomial distribution, the GLM version of edgeR is most closely related to the methods of this article in that it allows gene-specific dispersion estimates to vary around a central estimated trend and shares information across genes when estimating dispersions. The quasi-likelihood methods proposed in this article provide the additional advantages of incorporating uncertainty in estimated variances when testing for differential expression and providing a self-tuning approach to shrinking gene-specific dispersion estimates.

3.2.2 QL Method

We begin fitting a quasi-likelihood model by specifying a model for the mean and, up to a multiplicative constant, the variance for each observation as a function of its mean. Let Y_{ijk} represent the observed count for gene k in replicate j ($j = 1,\ldots,J$) of treatment group i ($i = 1,\ldots,I$), and let c_{ij} represent a normalization factor for the overall number of reads from replicate j in treatment group i (e.g., we set c_{ij} as the 0.75 quantile of reads from replicate j in treatment group i as recommended by Bullard et al. (2010)). Let $\mathrm{E}(Y_{ijk}|c_{ij}) = \mu_{ijk}$ where $\mu_{ijk} = \lambda_{ik}c_{ij}$ and λ_{ik} represents the normalized expression level of gene k in treatment group i. In this framework, gene k is defined to be equivalently expressed (EE) across treatments i and i' if $\lambda_{ik} = \lambda_{i'k}$ and differentially expressed (DE) otherwise. More generally, we can model $\log(\mu_{ijk})$ as a known constant ($\log c_{ij}$) plus a linear function of covariates and treatment

effects. Such extensions are straightforward and are not considered here to simplify the presentation.

Fitting a quasi-likelihood model requires specifying the variance of observed values, up to a proportionality constant, as a function of the modeled means. That is, one assumes $\text{Var}(Y_{ijk}) = \Phi_k V_k(\mu_{ijk})$, where $V_k(\mu_{ijk})$ is fully specified by the user and Φ_k is an unknown dispersion parameter that will be estimated from the data. Tables of commonly used variance functions, $V(\mu)$, and their corresponding quasi-likelihood functions can be found in McCullagh (1983) and McCullagh and Nelder (1983). For RNA-Seq data, it seems most reasonable to use $V_k(\mu_{ijk}) = \mu_{ijk} + \omega_k \mu_{ijk}^2$ (based on the negative binomial distribution, with some specified value of ω_k) or $V_k(\mu_{ijk}) = \mu_{ijk}$ (based on the Poisson distribution). However, our suggested methods can be used with any variance function for which there exists a corresponding quasi-likelihood function, $\ell(\mu_{ijk}|y_{ijk})$, that satisfies

$$\frac{\partial \ell(\mu_{ijk}|y_{ijk})}{\partial \mu_{ijk}} = \frac{y_{ijk} - \mu_{ijk}}{V_k(\mu_{ijk})}.$$

Note that both Φ_k and ω_k are dispersion parameters; ω_k (referred to as negative binomial dispersion) is a parameter of the negative binomial distribution and Φ_k (referred to as quasi-likelihood dispersion) is a proportionality constant used in quasi-likelihood models. In a quasi-negative binomial model, both ω_k and Φ_k are used to model the variance of observations from gene k; i.e., $\text{Var}(Y_{ijk}) = \Phi_k \left(\mu_{ijk} + \omega_k \mu_{ijk}^2\right)$.

The use of a quasi-likelihood approach based on a negative binomial distribution may seem unnecessary, as the negative binomial distribution has two parameters and provides great flexibility in modeling mean-variance relationships. However, existing popular methods for detecting differential expression with RNA-Seq data based on the negative binomial distribution fail to adequately account for uncertainty in parameter estimates. The simulations in Section 3.4 demonstrate that these methods produce an over-abundance of small p-values from EE genes, relative to a uniform distribution, even for data simulated from negative binomial distributions. These non-uniform distributions of p-values from EE genes are shown to produce q-values that substantially underestimate false discovery rates (FDR). A negative binomial implementation of the quasi-likelihood methods, using negative binomial dispersion parameter estimates from the GLM implementation of edgeR (Robinson et al., 2010; McCarthy et al., 2012), however, was found to produce q-values that were far more accurate. Thus, an important benefit of using a quasi-likelihood approach based on a negative binomial distribution is not the additional flexibility in modeling variances but rather the incorporation of uncertainty in the modeled variances

via the estimated quasi-likelihood dispersion parameter.

For each of K genes, parameter values are estimated by maximizing

$$\ell_k(\hat{\boldsymbol{\mu}}_k|\boldsymbol{y}_k) = \sum_{i,j} \ell_k(\hat{\mu}_{ijk}|y_{ijk}),$$

where $\boldsymbol{y}_k = (y_{11k},\ldots,y_{IJk})'$ is the vector of observations from the kth gene across samples, $\boldsymbol{\mu}_k = (\mu_{11k},\ldots,\mu_{IJk})'$ is the vector of the corresponding means, and $\ell_k(\mu|y)$ is the quasi-likelihood function corresponding to the variance function chosen for gene k.

Conducting a hypothesis test for differential expression using the quasi-likelihood approach involves computing a quasi-likelihood ratio test statistic and estimating the dispersion parameter, Φ_k (the proportionality constant from the specified mean-variance relationship). The quasi-likelihood ratio test statistic is computed as

$$LRT_k = 2\left(\ell_k(\hat{\boldsymbol{\mu}}_k|\boldsymbol{y}_k) - \ell_k(\tilde{\boldsymbol{\mu}}_k|\boldsymbol{y}_k)\right), \tag{3.1}$$

where $\tilde{\mu}_{ijk}$ and $\hat{\mu}_{ijk}$ are the maximum quasi-likelihood estimates for μ_{ijk} under the null and alternative hypotheses, respectively. When the mean-variance function has been correctly specified, McCullagh (1983) shows that under the null hypothesis

$$LRT_k \sim \Phi_k \chi^2_q + O_p(n^{-1/2}), \tag{3.2}$$

where q is the difference between the dimensions of the full and null-constrained mean parameter spaces and n is the total number of samples.

The dispersion parameter, Φ_k, can be estimated as

$$\hat{\Phi}_k = \frac{2\left(\ell_k(\boldsymbol{y}_k|\boldsymbol{y}_k) - \ell_k(\hat{\boldsymbol{\mu}}_k|\boldsymbol{y}_k)\right)}{n-p}, \tag{3.3}$$

where p is the dimension of the full-model mean parameter space. This deviance based estimator of Φ_k is asymptotically independent of maximum likelihood estimates for the parameters used to model $\boldsymbol{\mu}_k$ (McCullagh, 1983). Although this estimator has a similar form to Equation 3.1, its asymptotic distribution does not follow from Equation 3.2 for as n tends to ∞, $n-p$ also tends to ∞, and the derivation of Equation 3.2 requires that q be finite. For distributions that are asymptotically normal, as $\mu \to \infty$, (including Poisson distributions, but not other negative binomial distributions) Tjur (1998) shows that as counts (rather than the number of samples, n) tend to $\infty, \hat{\Phi}_k \sim \Phi_k \chi^2_{n-p}$ by approximating

the quasi-likelihood models with nonlinear regression models. Tjur suggests comparing the test statistic

$$F_{QL} = \frac{LRT_k/q}{\hat{\Phi}_k}$$

to an F-distribution with q and $n-p$ degrees of freedom. We refer to this approach as QL for quasi-likelihood.

While other dispersion estimators have better understood asymptotic distributions, we originally chose Equation 3.3 due to its symmetry with Equation 3.1. The numerator of F_{QL} is twice the difference between quasi-likelihoods of the full and reduced models, divided by the difference between the dimensions of the unconstrained and null-constrained parameter spaces. That is, the numerator is an estimate of the average change in deviance per constrained parameter. The denominator of F_{QL} is the estimated dispersion and, when the suggested deviance estimator is used, is equal to twice the difference between quasi-likelihoods of the saturated and full models, divided by the residual degrees of freedom. That is, the denominator is an estimate of the average change in deviance per residual degree of freedom. F_{QL} thus provides the average number of residual degrees of freedom each parameter constrained by the null hypothesis is worth in terms of change in deviance. This is an exact parallel to the F-tests produced in standard ANOVA tables and, as the simulation studies described in Section 3.4 demonstrate, makes the QL method robust to model misspecification.

Among alternative dispersion estimators, the most popular may be Pearson's estimator,

$$\hat{\Phi}_k^{Pearson} = \frac{1}{n-p}\sum_{i=1}^{I}\sum_{j=1}^{J}\left(Y_{ijk} - \hat{\mathrm{E}}(Y_{ijk})\right)^2 / \widehat{\mathrm{Var}}(Y_{ijk}).$$

We examined the performance of our suggested methods using Pearson's dispersion estimator in place of the deviance estimator. In general, Pearson dispersion estimates tended to be smaller than the corresponding deviance based dispersion estimates, and using the Pearson estimates led to liberal results (i.e. over-abundance of small p-values from EE genes and q-values that underestimated empirical FDRs), particularly for the quasi-negative binomial methods. We therefore recommend the deviance dispersion estimator when using methods described in this paper.

3.2.3 QLShrink Method

It is common for $n-p$ to be small in RNA-Seq experiments, so the QL approach often can be substantially improved by sharing information across genes when estimating dispersion parameters.

We suggest adapting the method described in Smyth (2004) for estimating gene-specific error variances for multiple linear models. Our approach places a scaled-inverse χ^2 prior distribution with d_0 degrees of freedom and scaling factor Φ_0 on each gene's dispersion, such that

$$d_0\Phi_0/\Phi_k \sim \chi^2_{d_0}. \tag{3.4}$$

We further assume that

$$(n-p)\hat{\Phi}_k/\Phi_k|\Phi_k \sim \chi^2_{n-p}, \tag{3.5}$$

based on, but not theoretically justified by, Equations 3.2 and 3.3. These assumptions produce an inverse-gamma posterior distribution such that

$$1/\Phi_k|\hat{\Phi}_k \sim \text{gamma}\left[.5(d_0+n-p), .5\left(d_0\Phi_0+(n-p)\hat{\Phi}_k\right)\right].$$

The hyperparameters d_0 and Φ_0 can be estimated from the distribution of $\hat{\Phi}_k$ using a method of moments approach described by Smyth (2004). A natural estimator of Φ_k can be formed by using the estimated posterior expectation as follows:

$$\hat{\Phi}^s_k = \hat{\text{E}}^{-1}(\Phi_k^{-1}|\hat{\Phi}_k) = \frac{\hat{d_0}\hat{\Phi}_0+(n-p)\hat{\Phi}_k}{\hat{d_0}+(n-p)}. \tag{3.6}$$

We compare the test statistic $LRT_k/(q\hat{\Phi}^s_k)$ to an F-distribution with q and $\hat{d_0}+n-p$ degrees of freedom. Given that Marioni et al. (2008) showed that variability among technical replicates is consistent with a Poisson model, we do not expect RNA-Seq data from biological replicates to be underdispersed. Thus, when using a quasi-Poisson model, we suggest using $\tilde{\Phi}^s_k = \max(1, \hat{\Phi}^s_k)$ as an estimator of Φ_k and comparing the test statistic $LRT_k/(q\tilde{\Phi}^s_k)$ to an F-distribution with q and $\hat{d_0}+n-p$ degrees of freedom. We refer to this approach as QLShrink.

3.2.4 QLSpline Method

A clear relationship is often present between estimated dispersions and average counts. (See Figure 3.1, for example.) In this scenario, it is beneficial to define a prior scaling factor, Φ_{0k}, for each gene as a function of the gene's average count. We recommend fitting a cubic spline to $\log(\hat{\Phi}_k)$ versus $\log(\bar{y}_{\cdot\cdot k})$, using cross-validation to determine the appropriate degrees of freedom to allow when fitting the spline.

Let $S_0(\cdot)$ be the resulting continuous function, and let $\hat{\Phi}_{0k} = \exp[S_0(\log \bar{y}_{\cdot\cdot k})]$. Under the assumption that the distribution of $\Phi_k|\hat{\Phi}_{0k}$ is defined by

$$d_0'\hat{\Phi}_{0k}/\Phi_k|\hat{\Phi}_{0k} \sim \chi^2_{d_0'}$$

and that Equation 3.5 holds, the ratio $\hat{\Phi}_k/\hat{\Phi}_{0k}|\hat{\Phi}_{0k}$ follows an F-distribution with $n-p$ and d_0' degrees of freedom for all k. When the cubic spline is fit on the log scale, we recommend allowing added flexibility by assuming $\hat{\Phi}_k/\hat{\Phi}_{0k}|\hat{\Phi}_{0k}$ follows a scaled F-distribution, with scaling factor γ. We then apply Smyth's method of moments approach to the set $\{\hat{\Phi}_k/\hat{\Phi}_{0k}\}_{k=1}^K$ to obtain estimates $\hat{d_0'}$ and $\hat{\gamma}$. Our suggested estimator for the kth gene's dispersion is

$$\hat{\Phi}_k^{(spline)} = \frac{\hat{d_0'}\hat{\Phi}_{0k}\hat{\gamma} + (n-p)\hat{\Phi}_k}{\hat{d_0'} + (n-p)}. \tag{3.7}$$

Fitting the cubic spline to $\log\left(\hat{\Phi}_k\right)$, as opposed to $\hat{\Phi}_k$, reduces the influence of extreme estimates on the spline fit but also produces estimates $\hat{\Phi}_{0k}$ that are too small. The additional scaling factor γ serves as a correction for using the log-scale and is strongly recommended by the authors. Fixing $\hat{\gamma} = 1$ in Equation 3.7 causes methods using the estimator to produce liberal results, particularly for small sample sizes. (e.g. For simulations with total sample sizes less than six, $\hat{\gamma}$ was often around 1.5.)

This estimation procedure shrinks $\hat{\Phi}_k$ toward $\hat{\Phi}_{0k}\hat{\gamma}$, which is a scale-adjusted, spline-based estimate of Φ_k. The extent of shrinkage depends on $\hat{d_0'}$ relative to $n-p$. As the degree of scatter around the spline fit (like that in Figure 3.1) decreases, $\hat{d_0'}$ increases and $\hat{\Phi}_{0k}\hat{\gamma}$ is more heavily weighted in $\hat{\Phi}_k^{(spline)}$. Conversely, as the scatter around the spline fit increases or as $n-p$ increases, the dispersion estimate based on the data for the kth gene, $\hat{\Phi}_k$, is more heavily weighted in $\hat{\Phi}_k^{(spline)}$. We then compare $LRT_k/(q\hat{\Phi}_k^{(spline)})$ to an F-distribution with q and $\hat{d_0'}+n-p$ degrees of freedom. As before, when using a quasi-Poisson model, we recommend letting $\tilde{\Phi}_k^{(spline)} = \max(1, \hat{\Phi}_k^{(spline)})$ and comparing $LRT_k/(q\tilde{\Phi}_k^{(spline)})$ to an F-distribution with q and $\hat{d_0'}+n-p$ degrees of freedom. We refer to this approach as QLSpline.

For this article, we consider Poisson and negative binomial implementations of the QL, QLShrink and QLSpline methods and use prefixes "Pois" and "NegBin" to denote which distribution was used when discussing results. Using a quasi-negative binomial model requires providing the negative binomial dispersion parameter ω_k in the equation $\text{Var}(Y_{ijk}) \propto \mu_{ijk} + \omega_k\mu_{ijk}^2$. For this paper, we provide estimates obtained from edgeR (Robinson et al., 2010) using the 'estimateGLMTrendedDisp' (McCarthy

et al., 2012) function. We use the default settings of this function, except when analyzing simulated data, for which we use min.n=100 in order to provide more points for edgeR to use when identifying a trend between the negative binomial dispersion estimates and average simulated counts.

3.3 Data Analysis

3.3.1 Fly Embryo Dataset

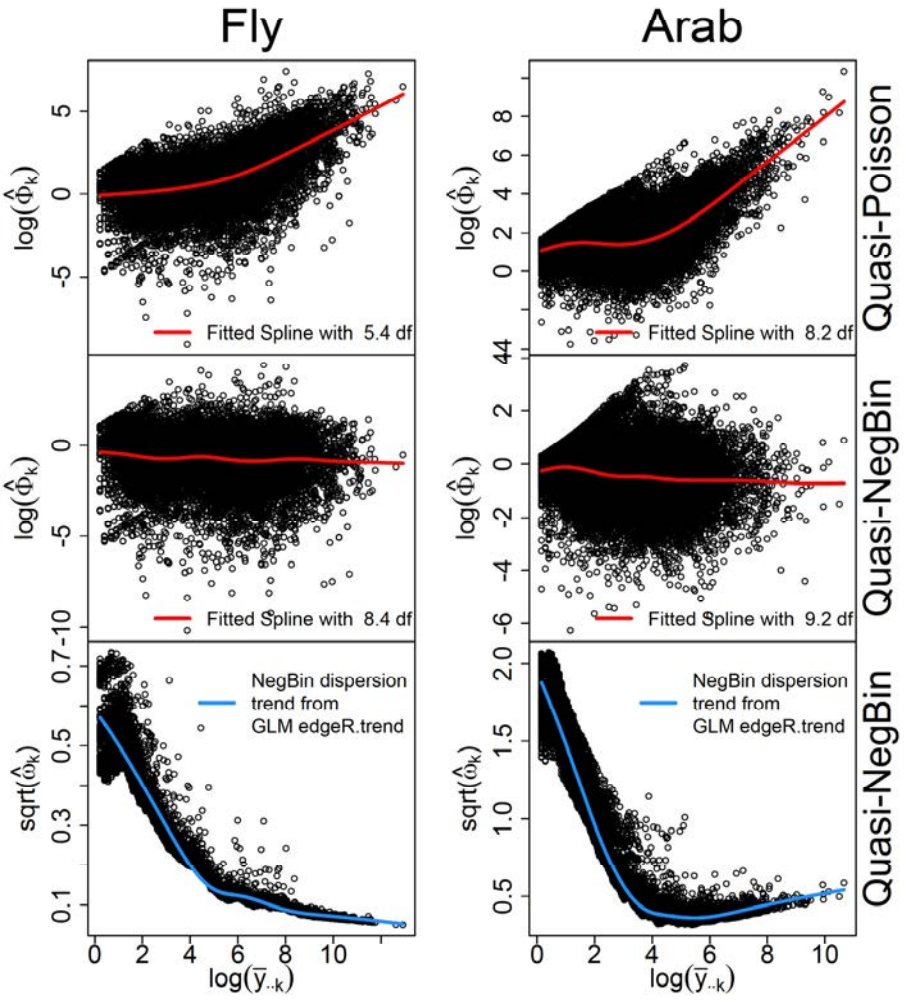

Figure 3.1 Estimated quasi-dispersions $\left(\hat{\Phi}_k\right)$ from quasi-Poisson (top) and quasi-negative binomial (middle) models versus average count with fitted splines for fly embryo (left) and Arabidopsis (right) data. Estimated dispersions for negative binomial distribution $(\hat{\omega}_k)$ from GLM edgeR.trend are shown in bottom row.

We first examine the fly embryo dataset provided in Anders and Huber (2010) from RNA-Seq experiments on fly embryos conducted by B. Wilczynski, Y.-H. Liu, N. Delhomme, and E. Furlong. The dataset includes count data for two biological replicates in each of two treatment groups labeled A and B, respectively. The left side of Figure 3.1 contains a scatterplot of the estimated quasi-Poisson and quasi-NegBin dispersions versus the average count for each gene for these data, along with the corresponding fitted cubic-splines used in the QLSpline methods. There is little relation between quasi-likelihood dispersion estimates, $\hat{\Phi}_k$, and the average count for the quasi-NegBin model, which is not surprising because negative binomial dispersion parameter estimates, $\hat{\omega}_k$, provided by edgeR have already undergone trend-based shrinkage, also shown in Figure 3.1.

The dataset contains 13230 genes for which there were at least two samples with non-zero counts and at least five total counts combined across the four samples. We tested each of these genes for differential expression between groups A and B with the following methods: DESeq (Anders and Huber, 2010), TSPM (Auer and Doerge, 2011), NBPSeq (Di et al., 2011), six implementations of edgeR (Robinson et al., 2010) formed by factorial combinations of testing procedure (exact (Robinson and Smyth, 2007, 2008) or GLM (McCarthy et al., 2012)) and dispersion estimation method (common dispersion [com], non-trended tagwise [tgw], or trended tagwise [trend]), and the QL, QLShrink and QLSpline methods applied to quasi-Poisson and quasi-negative binomial models. For each method, its recommended approach was used to account for differences in library sizes. The QL method group and TSPM used the 0.75 quantile of the read count distribution from each sample, as recommended by Bullard et al. (2010).

The analyses in this report used the following R packages to implement their corresponding methods: DESeq (version 1.6.1), edgeR (version 2.4.3) and NBPSeq (version 0.1.4). Code for implementing the TSPM method was taken from the website provided by Auer and Doerge (2011). Unless otherwise noted, the default settings for these packages were used during analysis.

Analysis results from the fly embryo data are summarized in the left side of Figure 3.2. For each method, we assigned p-values to bins of width 0.05 and used the number of p-values assigned to each bin to construct histogram curves. We applied the method of Nettleton et al. (2006) to the distribution of p-values resulting from the application of each method in order to obtain q-values and estimates of the total number of DE genes. The methods produced drastically different estimates of the total number

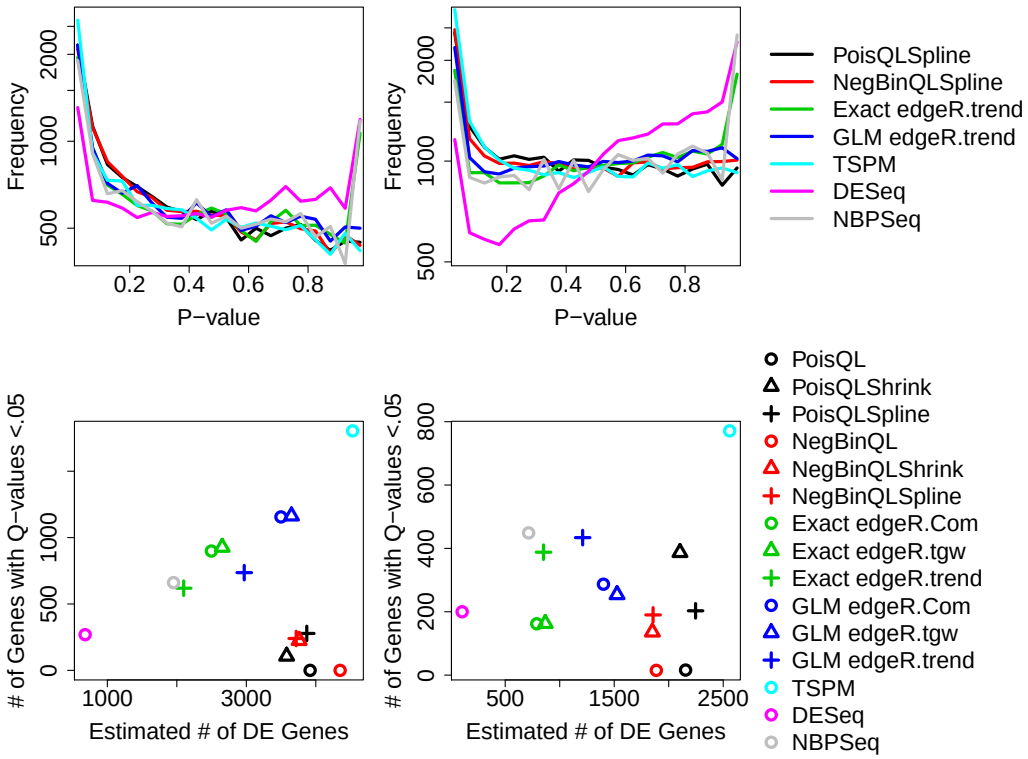

Figure 3.2 Histograms of p-values (top) and number of genes with q-values less than 0.05 versus estimated number of DE genes (bottom) for fly embryo (left) and Arabidopsis (right) data.

of DE genes (from 681 to 4530) and the number of genes with q-values less than .05 (from 0 to 1804). The p-value histograms for methods that used the exact test of Robinson and Smyth (2007) (i.e. exact edgeR, DESeq and NBPSeq) exhibited a spike for large p-values, which led to conservative estimates of the total number of DE genes.

The scatterplots in Figure 3.2 are primarily intended to display the large differences between the results from the considered methods. By themselves, these results do not provide sufficient information to evaluate each method. Generally speaking, the method with greatest power to detect differential expression is preferred, so long as the method allows researchers to accurately estimate or control false discovery rates. It is not possible to assess the performance of error rate control or estimation when the true status (EE or DE) of each analyzed gene is unknown, which is why we evaluate method performance through simulation studies.

In most cases when the goal of analyzing RNA-Seq data is to identify DE genes, resource constraints

Table 3.1 Overlap in methods' lists of top 200 genes for fly embryo (top) and Arabidopsis (bottom) data.

Method	1	2	3	4	5	6
PoisQLSpline (1)	200					
NegBinQLSpline (2)	189	200				
Exact edgeR.trend (3)	172	172	200			
GLM edgeR.trend (4)	175	175	183	200		
TSPM (5)	77	77	68	67	200	
DESeq (6)	168	161	152	152	81	200
NBPSeq (7)	153	151	161	167	50	133
Method	1	2	3	4	5	6
PoisQLSpline (1)	200					
NegBinQLSpline (2)	177	200				
Exact edgeR.trend (3)	158	160	200			
GLM edgeR.trend (4)	160	160	187	200		
TSPM (5)	25	26	14	15	200	
DESeq (6)	164	157	160	154	12	200
NBPSeq (7)	100	105	124	113	0	113

limit the number of genes that researchers will follow up with further study. Thus, overlap between lists from each method containing a fixed number of the most significant genes is an important feature for assessing similarity between methods' results. The top half of Table 3.1 provides the size of pairwise intersections of lists containing the 200 most significant genes from each of seven methods.

3.3.2 Arabidopsis Dataset

We also examined the Arabidopsis dataset provided as "arab" in the R package NBPSeq (Di et al., 2011). The dataset includes count data for three biological replicates in each of two treatments in which leaves were inoculated with either a *Pseudomonas syringae* DC3000 mutant bacteria strain or a mock inoculant. The right side of Figure 3.1 contains a scatterplot of the estimated quasi-Poisson and quasi-NegBin dispersions versus the average count for each gene for these data, along with the corresponding fitted cubic-splines used in the QLSpline methods. The dataset contains 21185 genes for which there were at least two samples with non-zero counts and at least seven total counts combined across the six samples. We tested each of these genes for differential expression between two treatment

conditions with the same methods used to analyze the fly embryo dataset. Code and corresponding output for implementing the PoisQL, PoisQLShrink and PoisQLSpline methods for these data via the R (R Development Core Team, 2011) package QuasiSeq is shown in Section 3.6.1.

The right side of Figure 3.2 summarizes analysis results from the Arabidopsis dataset when assuming a completely randomized experimental design (i.e. no replicate effects), as was done in Di et al. (2011). The methods produced drastically different estimates of the total number of DE genes (from 105 to 2559) and the number of genes with q-values less than 0.05 (from 15 to 771). The p-value histogram for DESeq was severely J-shaped, and NBPSeq and exact edgeR again exhibited a spike for large p-values, which led to conservative estimates of the total number of DE genes. The bottom half of Table 3.1 provides the size of pairwise intersections of lists containing the 200 most significant genes from each of seven methods.

Describing the experiment behind the Arabidopsis dataset, Cumbie et al. (2011) writes, "Each treatment was done as biological triplicates with each pair of replicates done at separate times..." This description suggests that block effects should be included when analyzing these data, unless there is evidence that block effects are insignificant. The exact test of Robinson and Smyth (2007) examines differences between two levels of a common factor and does not accommodate nuisance factors, so the exact edgeR, NBPSeq and DESeq methods are unable to incorporate (or test for) block effects. The TSPM, GLM edgeR and QL methods are all built from GLMs and can accommodate nuisance factors by using an appropriate design matrix when estimating parameters.

When block effects are included in the model, estimating the variance for a gene in a reasonable manner requires having at least three total samples that have non-zero counts, with at least one of those samples coming from each treatment group. (Otherwise, the full model provides the same fitted values as the saturated model, so the variance is estimated to be essentially zero.) We analyzed the 20224 genes contained in the Arabidopsis data that met this criteria and that had at least seven total counts across all six samples. Figure 3.3 provides estimates of the total number of non-null genes and the numbers of genes with q-values less than 0.05 resulting from tests for block and treatment effects, respectively, in the Arabidopsis data. These results provide strong evidence that block effects are present and that incorporating block effects significantly improves power to detect differential expression between treatments for these data.

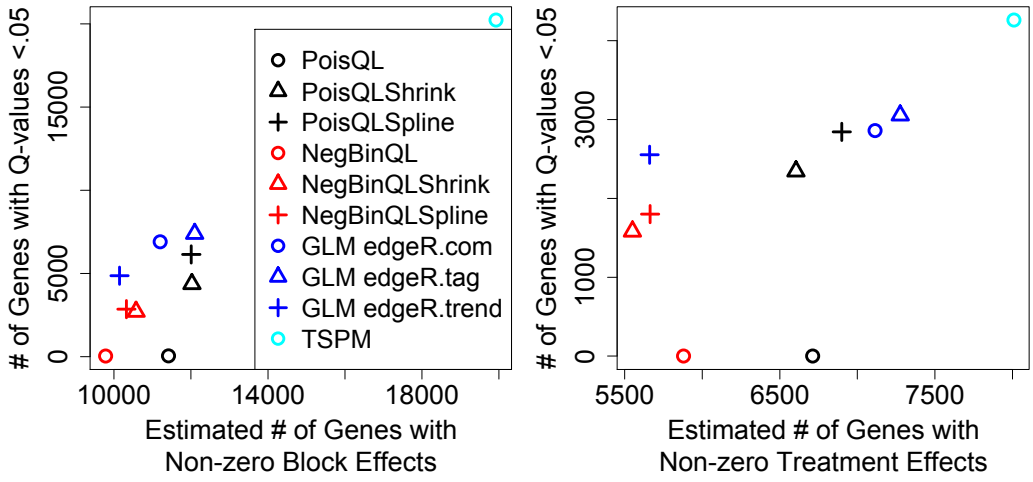

Figure 3.3 Number of genes with q-values less than 0.05 versus estimated number of non-null genes based on p-values testing for presence of block effects (left) and treatment effects (right) for Arabidopsis data.

3.4 Simulation Study

3.4.1 Simulation Descriptions

To examine the effectiveness of our suggested approach, we conducted a series of simulations for sample sizes of 4, 6 and 10, split evenly between two treatment groups. To facilitate comparison with other methods, simulated genes with average counts less than 1 or more than 1,000,000 total counts were replaced with new simulated data before analyzing. The former are genes whose count data contain little or no information about differential expression that can be detected with any method. The latter represents genes with high counts that caused computational problems for a competing method. Each simulation scenario was repeated 200 times, and each dataset contained simulated counts for 1000 DE and 4000 EE genes.

3.4.1.1 Negative Binomial Simulations

We simulated negative binomial data using parameters guided by sample averages and dispersion estimates from the fly embryo and Arabidopsis datasets. For the fly embryo and Arabidopsis dataset, let $\bar{y}_{\cdot\cdot k}$ denote the sample average of the four and six observations, respectively, from gene k. Let $\hat{\omega}_k$

denote the estimated parameter for the negative binomial variance function $\left(\text{Var}(Y_{ijk}) = \mu_{ijk} + \omega_k \mu_{ijk}^2\right)$ obtained from the edgeR exact test tagwise dispersion estimation procedure with the trend option and a prior.n specification of 1. Figure 3.4 displays plots of $\hat{\omega}_k$ versus $\bar{y}_{\cdot\cdot k}$ that were used in these simulations.

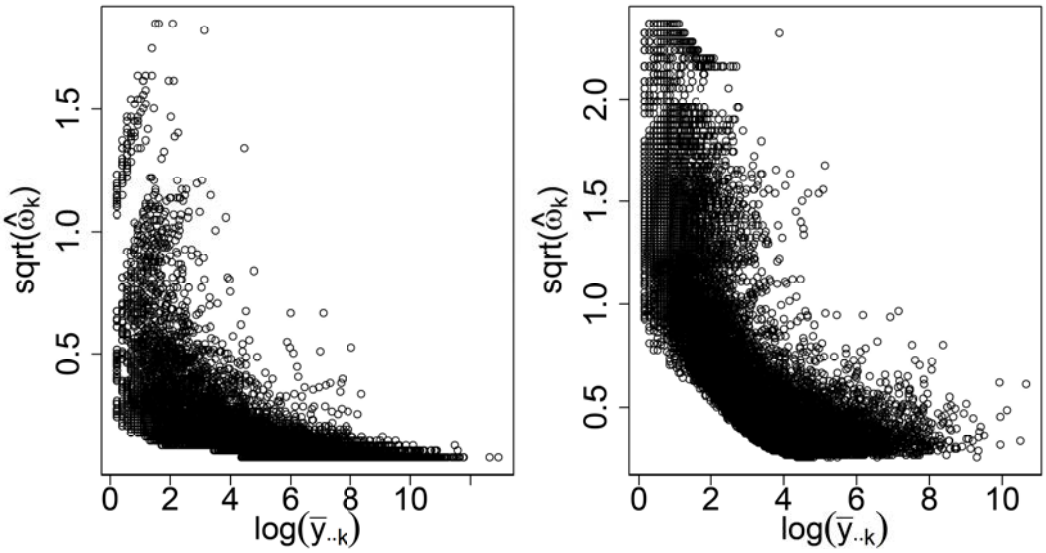

Figure 3.4 Sample averages and negative binomial dispersion parameter estimates used for simulations based on fly embryo (left) and Arabidopsis (right) data.

Data were simulated from negative binomial distributions for the kth gene in the following manner. Let k' index a gene randomly selected from the real dataset. If the kth simulated gene was to be EE, we let $\lambda_{ik} = \bar{y}_{\cdot\cdot k'}$ for $i = 1, 2$. If the kth simulated gene was to be DE, for simulations based on the fly embryo data, we sampled a fold change factor, B_k, in the following manner. We set $B_k = B_{k1} + B_{k2}$, where B_{k1} was sampled from an inverse-gamma distribution with rate parameter 1 and shape parameter $S\bar{y}_{\cdot\cdot k'}^{1/8}$ and B_{k2} was sampled from a uniform distribution with endpoints L and U. (Values for L and U are provided in Table 3.2. We adjusted the severity of simulated fold changes to maintain moderate separation of EE and DE genes by using $S = 1, 1.2, 1.5$ for $n = 4, 6, 10$, respectively.) For simulations based on the Arabidopsis data, B_{k1} was sampled from an inverse-gamma distribution with rate parameter 1 and shape parameter $S\log\left(\bar{y}_{\cdot\cdot k'}\right)^{1/8}$. Small and large expression levels of $\bar{y}_{\cdot\cdot k'}/\sqrt{B_k}$ and $\bar{y}_{\cdot\cdot k'}\sqrt{B_k} + 5$, respectively, were randomly assigned between λ_{1k} and λ_{2k}. Library size factors were simulated according to $\log_2 c_{ij} \sim \text{Normal}(0, 0.5^2)$, where c_{ij} is the simulated library size factor for replicate j in treatment i. Final counts were simulated from a negative binomial distribution with mean $\mu_{ijk} = \lambda_{ik} c_{ij}$

and variance $\mu_{ijk} + \hat{\omega}_{k'}\mu_{ijk}^2$.

The techniques for simulating fold changes were chosen to reproduce the relationship between estimated fold change and average count seen in the fly embryo and Arabidopsis data for the $n = 4$ and $n = 6$ simulations, respectively. (See Figure 3.5.)

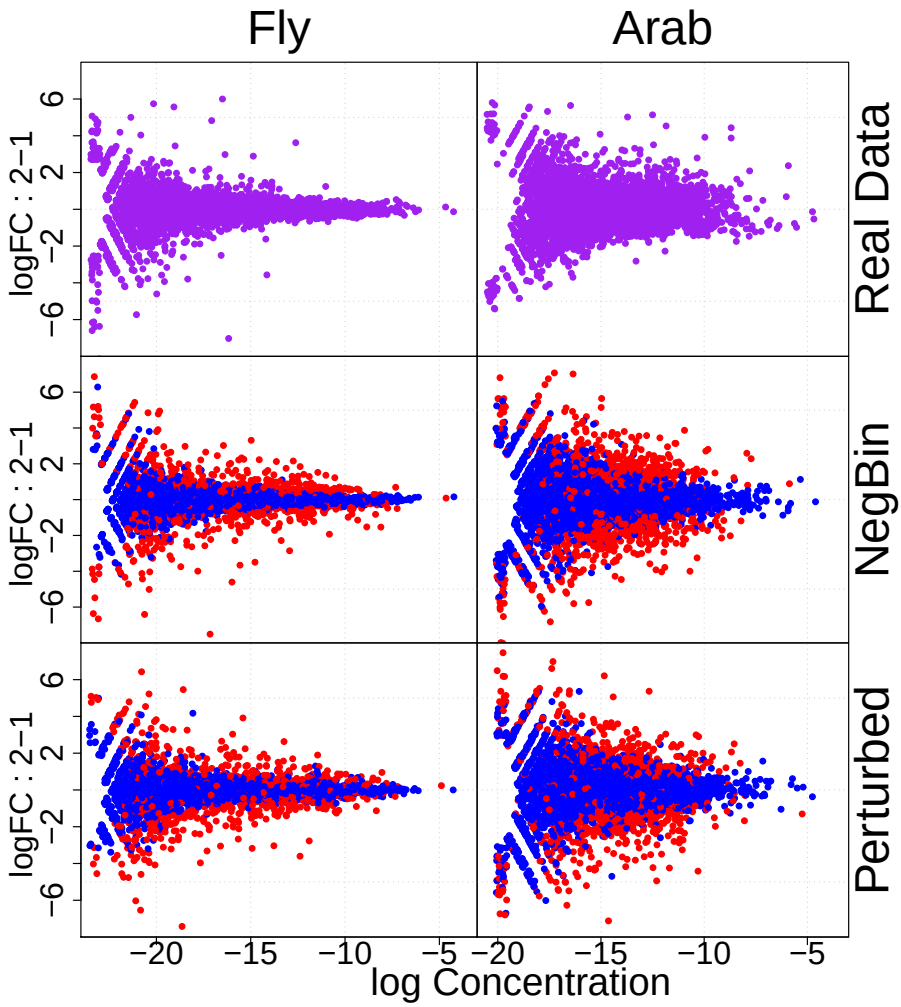

Figure 3.5 Estimated log fold change versus log average count for actual (top), negative binomial simulated (middle) and perturbed simulated (bottom) data from fly embryo (left, $n = 4$) and Arabidopsis (right, $n = 6$) datasets. For simulated datasets, DE and EE genes are marked with red and blue dots, respectively, and library size factors were simulated according to $\log_2 c_{ij} \sim \text{Normal}(0, 0.5^2)$.

In each of the considered methods, variance is modeled using (gene-specific) functions of the mean. Regardless of what mean-variance relationship is assumed, within reason, observations within a gene

Table 3.2 Parameters used to simulate fold changes.

Model	Data Set	L	U
NegBin	Fly	0.25	0.75
NegBin	Arab	1.25	1.75
Perturbed	Fly	0.25	0.75
Perturbed	Arab	1	1.5

with roughly the same modeled means will have roughly the same modeled variance. Effects of experimental design factors (e.g. treatment or blocking factors) and differences between library sizes produce differences among modeled means for observations from a single gene. When library sizes are roughly equal, the chosen mean-variance relationship has a limited effect on method performance. When library sizes differ greatly, even replicate observations from the same gene and treatment can have very different means, and the specified mean-variance relationship can strongly impact method performance. For this reason, we repeat the negative binomial simulations using assigned library size factors of $c_{ij} = 0.3$ or $c_{ij} = 3$, alternating between every other sample. These simulations are referred to as "10-fold NegBin."

To examine method sensitivity to the data-generating model, we also simulated data from slight perturbations of negative binomial distributions using parameters guided by sample averages and dispersion estimates from the fly embryo and Arabidopsis datasets. These simulations began by sampling a mean and dispersion pair from the real data set $(\bar{y}_{\cdot\cdot k'}, \hat{\omega}_k)$, using assigned library size factors alternating between $c_{ij} = 0.3$ and $c_{ij} = 3$ and, for DE genes, generating a fold change factor, B_k, in exactly the same way as was done in the negative binomial simulations, using the parameter values given in Table 3.2. Let $\lambda'_{ijk} = \bar{y}_{\cdot\cdot k'} c_{ij}$ if gene k was simulated as EE and let $\lambda'_{ijk} = \bar{y}_{\cdot\cdot k'} c_{ij} / \sqrt{B_k}$ (or $\bar{y}_{\cdot\cdot k'} c_{ij} \sqrt{B_k} + 5$) if gene k was simulated as DE.

To modify the data-generating model, we generated a perturbation effect, $\varsigma_k \sim \text{Normal}(0,\ 0.1)$, and simulated means λ_{ijk} from a gamma distribution with shape parameter ${\lambda'_{ijk}}^{\varsigma_k} / \hat{\omega}_k$ and rate parameter ${\lambda'_{ijk}}^{\varsigma_k - 1} / \hat{\omega}_k$. Final counts were simulated as $Y^{sim}_{ijk} | \lambda_{ijk} \sim \text{Poisson}(\lambda_{ijk})$. The final counts have conditional mean and variance $\text{E}(Y^{sim}_{ijk} | \lambda'_{ijk}, \varsigma_k) = \lambda'_{ijk}$ and $\text{Var}(Y^{sim}_{ijk} | \lambda'_{ijk}, \varsigma_k) = \lambda'_{ijk} + \hat{\omega}_k {\lambda'_{ijk}}^{2-\varsigma_k}$, which is a slight variation from the mean-variance relationship of the negative binomial distribution. We refer to these simulations as "10-fold perturbed."

3.4.2 Simulation Results

We evaluated each method's performance according to two criteria: separation of DE and EE genes in significance rankings as seen in discovery versus false discovery curves and uniformity of the empirical distribution of p-values coming from EE genes. We also observed the effect that non-uniform null p-value distributions can have on estimated false discovery rates by comparing empirical FDRs (eFDR) to q-values. We report simulation results through a combination of plots and tables. The plotted curves describe average behavior over 200 iterations for each simulation scenario. For each curve, solid thin lines located $\pm$ two standard errors around the mean are also included, providing approximate 95% pointwise confidence intervals, although most of the standard error lines have merged with their corresponding mean line.

We began our simulation study with every method whose results are reported for the fly embryo and Arabidopsis datasets. To control the number of results to report and to increase the speed of conducting simulations, we kept only the best performing methods from each of the following four classes: Poisson QL, negative binomial QL, GLM edgeR, and exact edgeR. Across most scenarios, the QLSpline method exhibited the best performance of the quasi-Poisson methods. Under a quasi-negative binomial model, the QLShrink and QLSpline methods performed similarly well. This was not surprising because only a slight relationship was present between quasi-likelihood dispersion estimates, $\hat{\Phi}_k$, and average counts for the quasi-NegBin model. We chose to include the QLSpline approach. The trend implementations of the exact test and GLM versions of edgeR generally outperformed their constant dispersion and non-trend tagwise dispersion counterparts. We also included results from TSPM, DESeq and NBPSeq.

The solid curves in Figure 3.6 display curves relating number of false discoveries to total number of discoveries for the $n = 6$ simulations. The qualitative traits of these curves were similar for other examined sample sizes. It is difficult to assess relative performance from these plots, although it is clear that the top five methods are GLM edgeR.trend, exact edgeR.trend, DESeq, PoisQLSpline, and NegBinQLSpline. To better examine the relative performance among the top five methods for each simulation scenario, we subtract the average number of discoveries (across 200 simulation iterations) for the PoisQLSpline method from the curve for each of the top five methods and plot the differences in Figures 3.7 through 3.9. Figure 3.7 shows that PoisQLSpline provided the best significance rankings

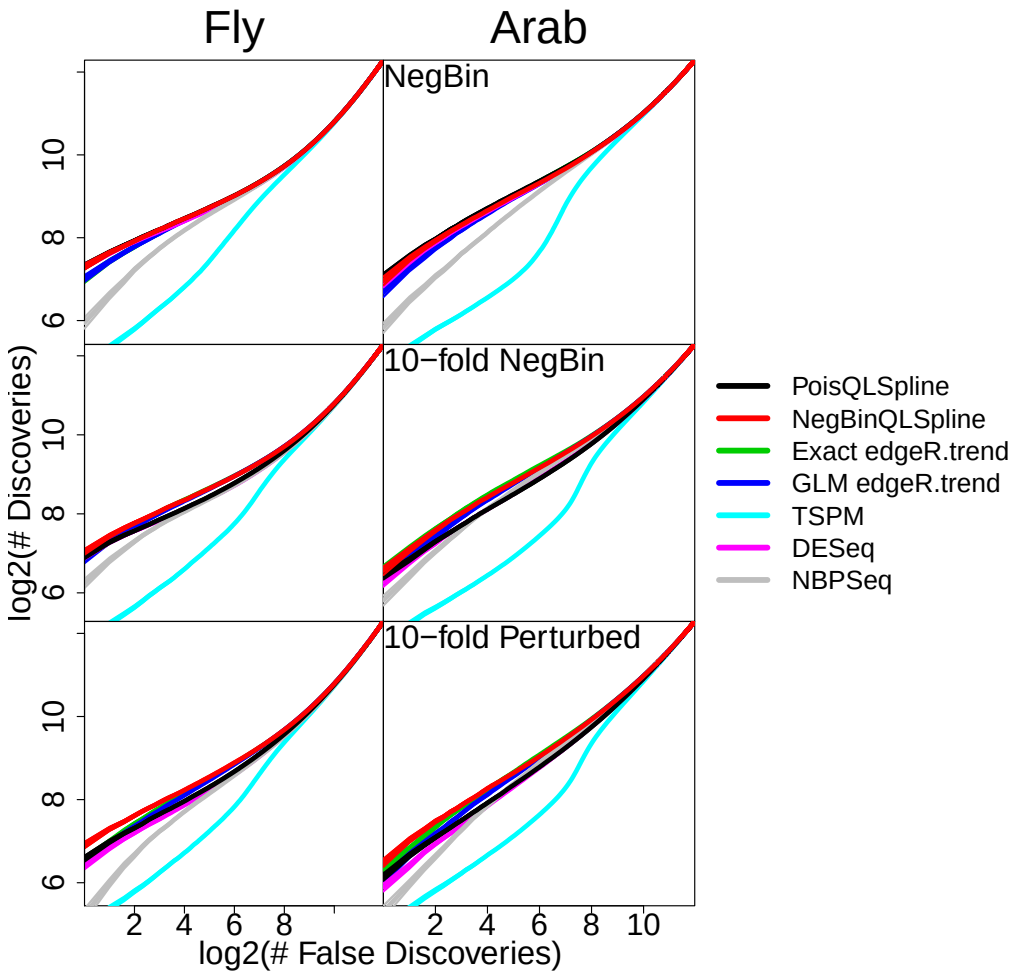

Figure 3.6 Curves relating average number of total discoveries to average number of false discoveries for negative binomial (top) and perturbed NegBin (bottom) simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 6$.

among the most significant genes in the simulations with moderate differences between library size factors. For simulation scenarios using 10-fold differences between library sizes, NegBinQLSpline and exact edgeR.trend were the top performers, although parameter estimates from exact edgeR.trend were used when simulating the data. As an example, in the $n = 10$ 10-fold perturbed simulations based on the fly embryo data, NegBinQLSpline identified between 25 and 75 more true positives than the other methods over a range of 0 to 10 false discoveries. Curves for PoisQLSpline and DESeq were far lower than the other three methods in simulations using 10-fold library size differences.

Improved significance rankings lead to fewer false positives (and more true positives) appearing on

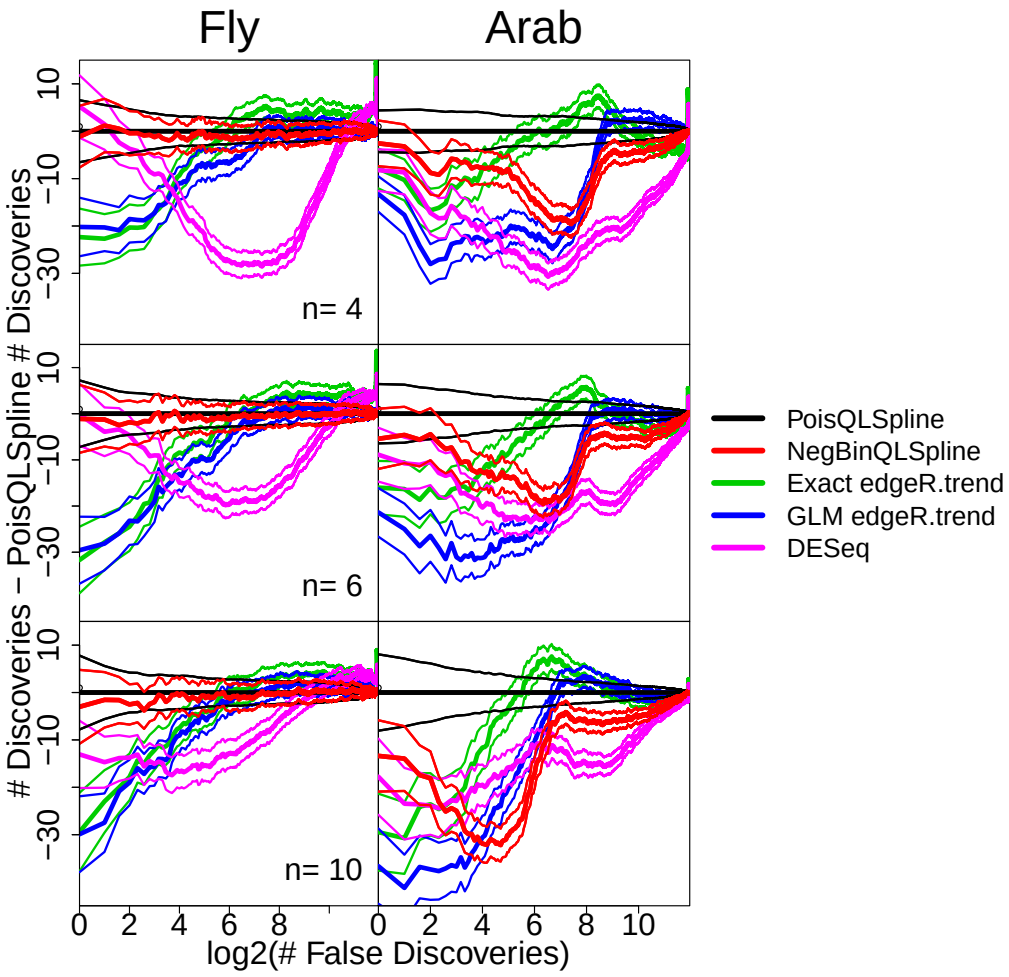

Figure 3.7 Curves relating difference in average number of total discoveries to average number of false discoveries for negative binomial simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

a list containing a fixed number of genes. This is important as resource constraints limit the number of genes that researchers can follow up on in future studies. To facilitate a direct comparison among the methods, the average number of DE genes in the 200 most significant genes for each method are provided in Tables 3.3-3.8. These numbers are useful for putting the power and sensitivity of the methods into a practical perspective. In the $n = 4$ 10-fold perturbed simulations based on the fly embryo dataset, for example, NegBinQLSpline averaged 190.6 DE genes while its closest competitor, Exact edgeR.trend, averaged 188.6 DE genes in their respective lists of 200 most significant genes. For simulations with moderate library size differences, the QLSpline methods had as many or more truly DE

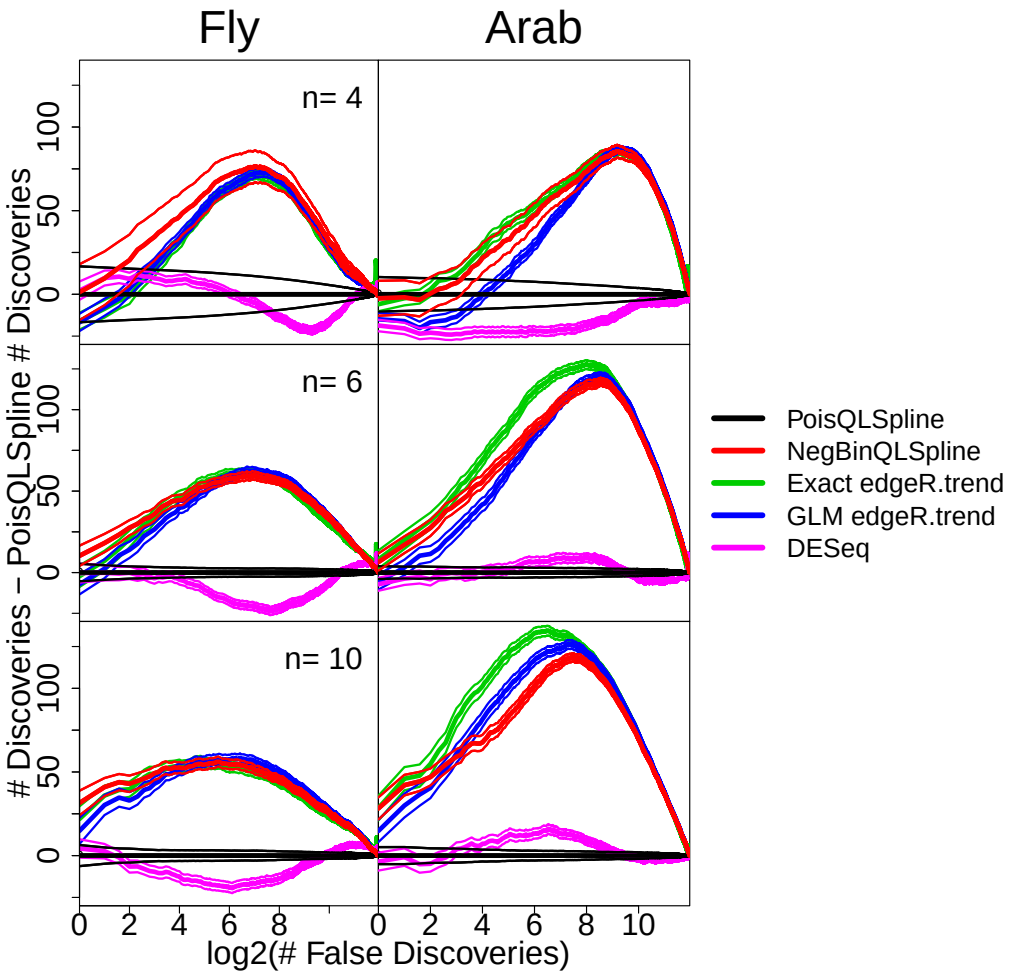

Figure 3.8 Curves relating difference in average number of total discoveries to average number of false discoveries for 10-fold NegBin simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

genes contained in the 200 most significant genes than each competing method. For simulations with 10-fold library size differences, NegBinQLSpline continued to perform well relative to its competitors, but the performance of PoisQLSpline dropped dramatically. Overall, these results demonstrate that NegBinQLSpline produced average significance rankings as good as or better than the other methods across a variety of simulations.

We next examine the distribution of p-values for simulated EE genes. For each method in each simulation, p-values from the 4000 EE genes were assigned to bins of width 0.005, and the number of p-values assigned to each bin was recorded. Figures 3.10 through 3.12 display histogram curves,

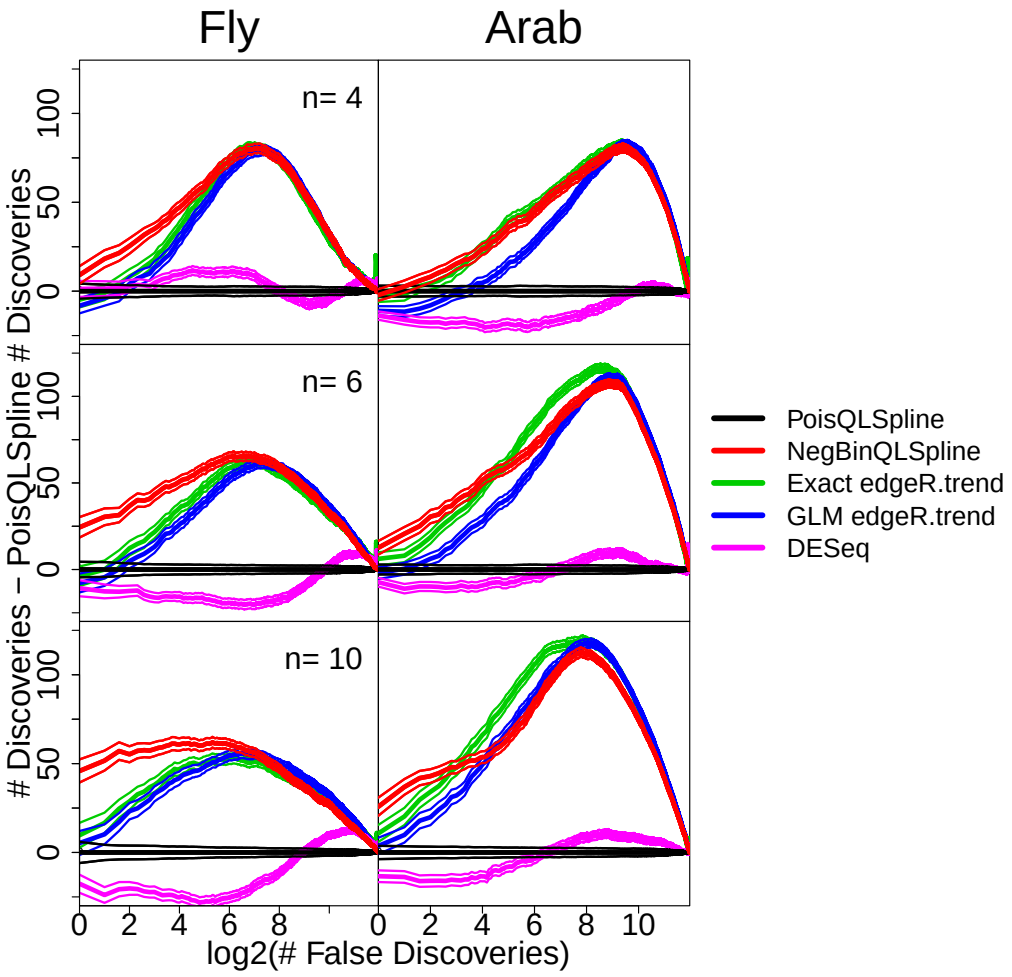

Figure 3.9 Curves relating difference in average number of total discoveries to average number of false discoveries for 10-fold perturbed simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

providing the average density of p-values assigned to each bin. The dashed orange line provides a reference for comparison with the uniform distribution. For the purposes of estimating false discovery rates, the most influential deviation from uniformity occurs when there are too many small p-values. The plots display the p-value axis on a log-scale in order to focus on the distribution of null p-values between 0 and 0.1.

The TSPM, NBPSeq, GLM edgeR.trend, and exact edgeR.trend methods display an over-abundance of small p-values relative to a uniform distribution in all simulation scenarios. This can be explained by the fact that the edgeR and NBPSeq methods do not account for uncertainty in their negative binomial

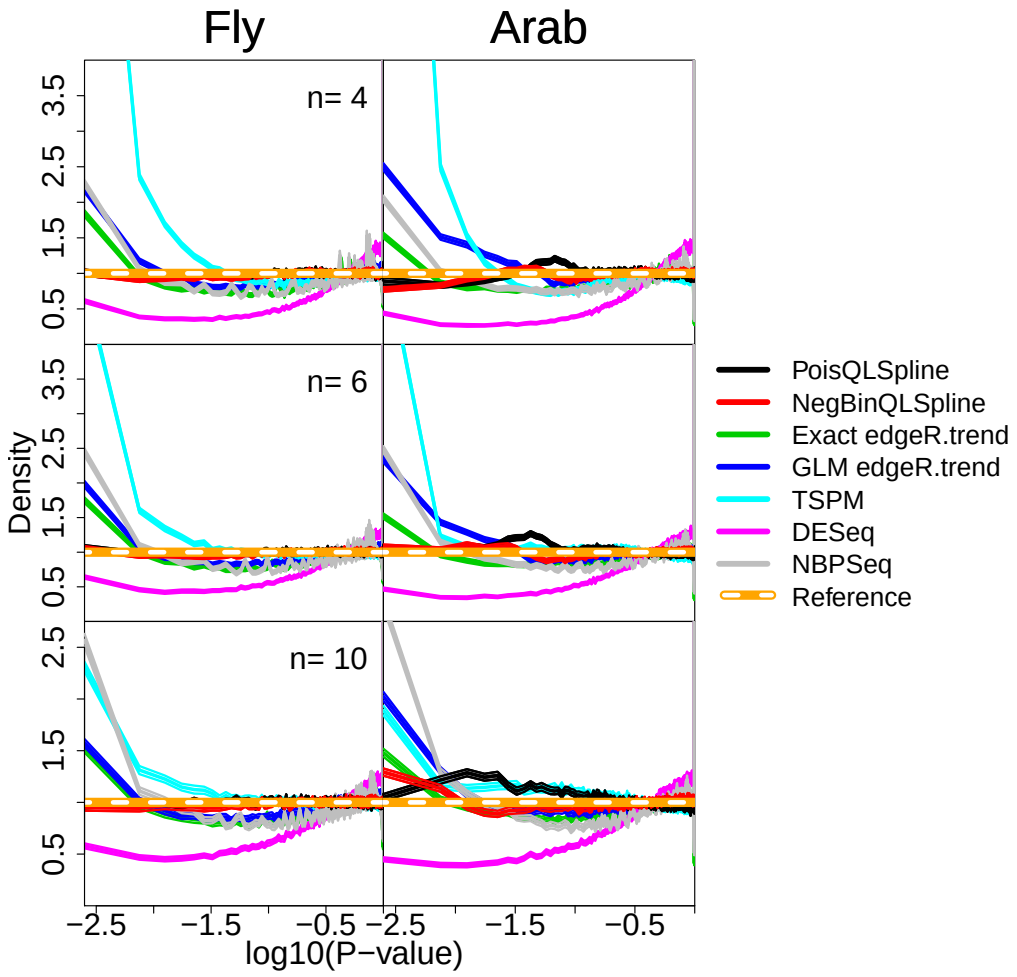

Figure 3.10 Histograms of p-values for EE genes in negative binomial simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

dispersion parameter estimates, and the TSPM method uses a Poisson-based approach that only adjusts for overdispersion for genes in which overdispersion is found to be statistically significant. Although DESeq also fails to account for uncertainty in its negative binomial dispersion parameter estimates, it generally produced strongly conservative results (i.e. small p-values are under-represented in the distributions of null p-values from DESeq). However, in nearly every simulation scenario there is a small peak in average density for the bin corresponding to p-values between 0 and 0.005 relative to the density of other bins corresponding to p-values less than 0.1. In simulations with moderate library size differences, the distributions of null p-values from the QLSpline methods show little deviation from uniformity as their curves are almost entirely hidden behind the dashed orange reference line. In

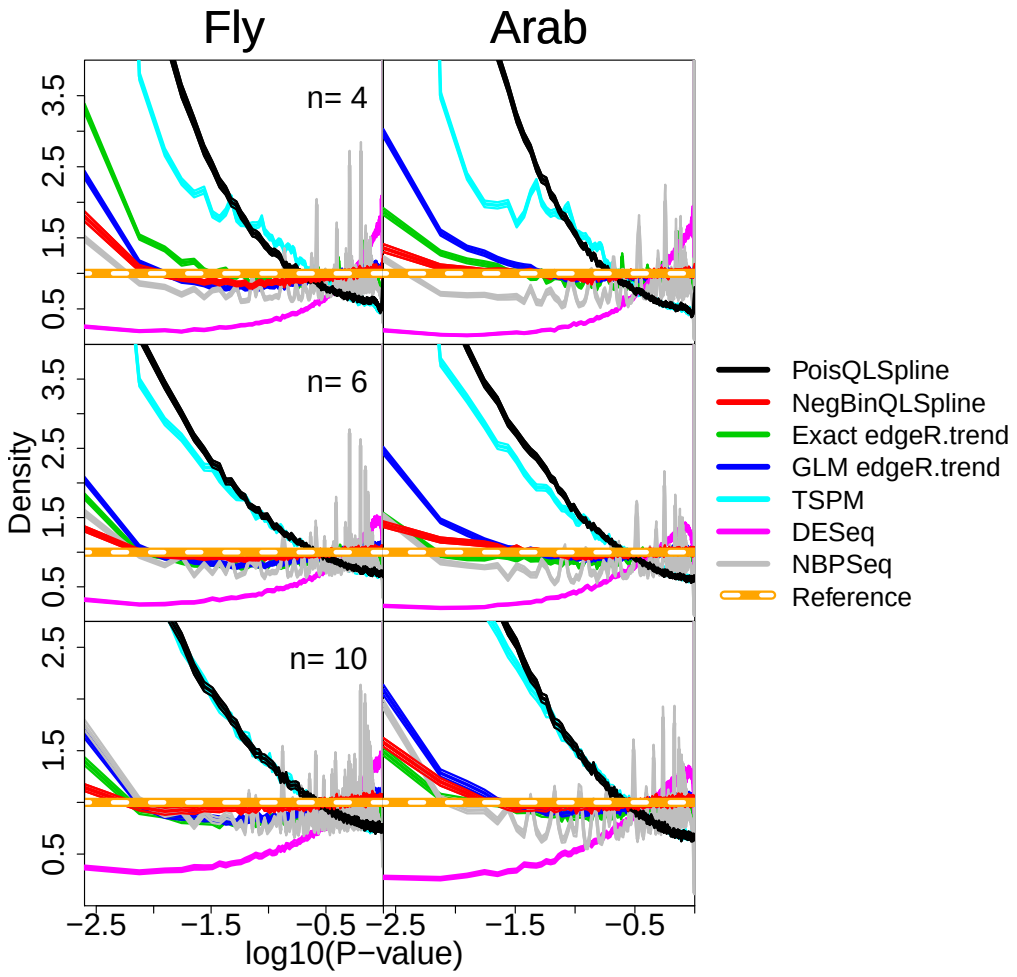

Figure 3.11 Histograms of p-values for EE genes in 10-fold NegBin simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

simulations with 10-fold library size differences, PoisQLSpline produced a severe over-abundance of small p-values. For NegBinQLSpline, small p-values were over and under represented in the 10-fold negative binomial and perturbed simulations, respectively.

A surplus of very small (<0.005) p-values can drastically affect false discovery rate estimates. As a demonstration, we compare empirical false discovery rates (eFDR) to q-values. The eFDR of gene k reports the proportion of genes that were EE from the set of genes that have p-values as small as or smaller than the p-value of gene k. Q-values are obtained by applying the method of Nettleton et al. (2006) to the distribution of p-values resulting from the application of each method. In this section we refer to methods as being liberal and conservative when their distributions of null p-values lead to

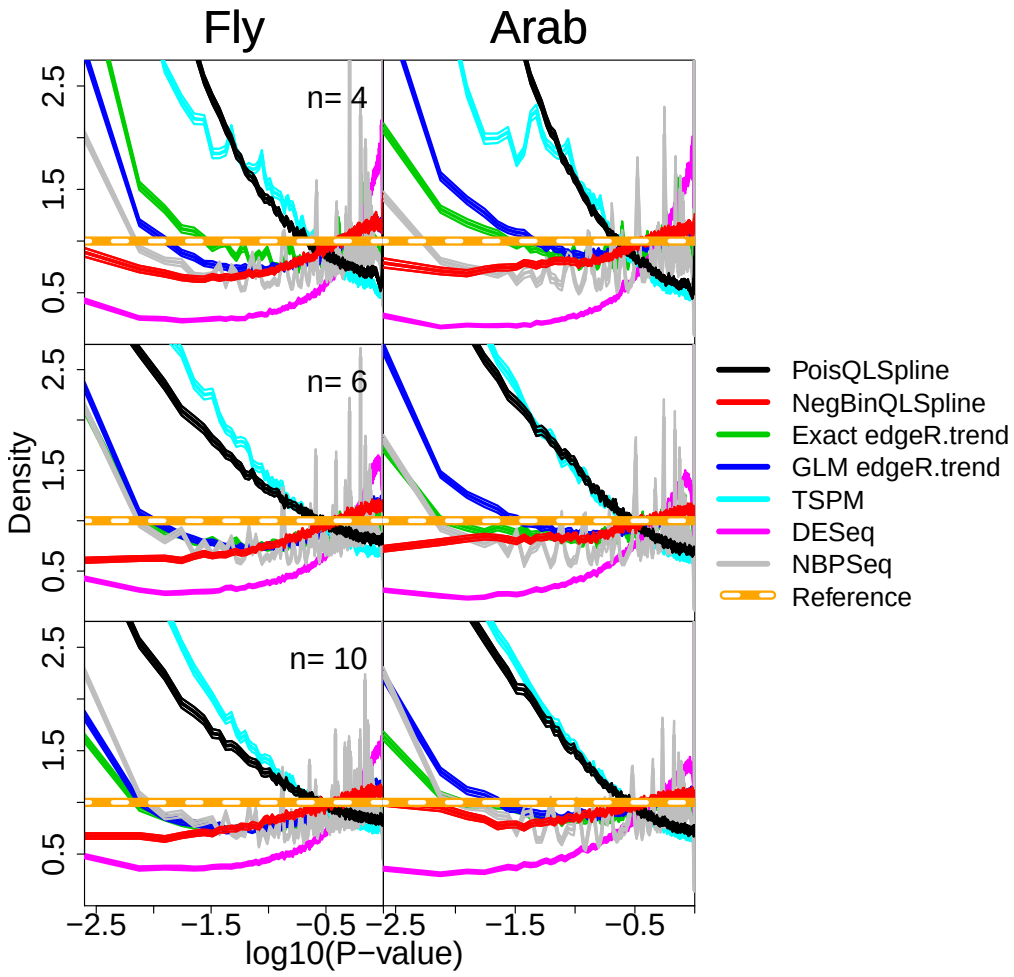

Figure 3.12 Histograms of p-values for EE genes in 10-fold perturbed simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

q-values that underestimate or overestimate FDRs, respectively. It should be noted that R packages for many competing methods include an approach, such as the Benjamini and Hochberg procedure, to control, rather than estimate, FDRs. We are not investigating the performance of FDR control approaches from each package, but examining the impact of non-uniform null p-values on q-values. In this sense, if a method is neither conservative nor liberal, then the q-value for any given gene should closely match its eFDR. For example, if the gene with the Mth smallest p-value has a corresponding q-value of .05, then roughly 5% of the M genes with p-values as small or smaller should be EE.

To examine if this characteristic held for each method, we plotted average eFDRs versus q-values for each scenario. The solid curves in Figures 3.13 through 3.15 display curves from the negative binomial,

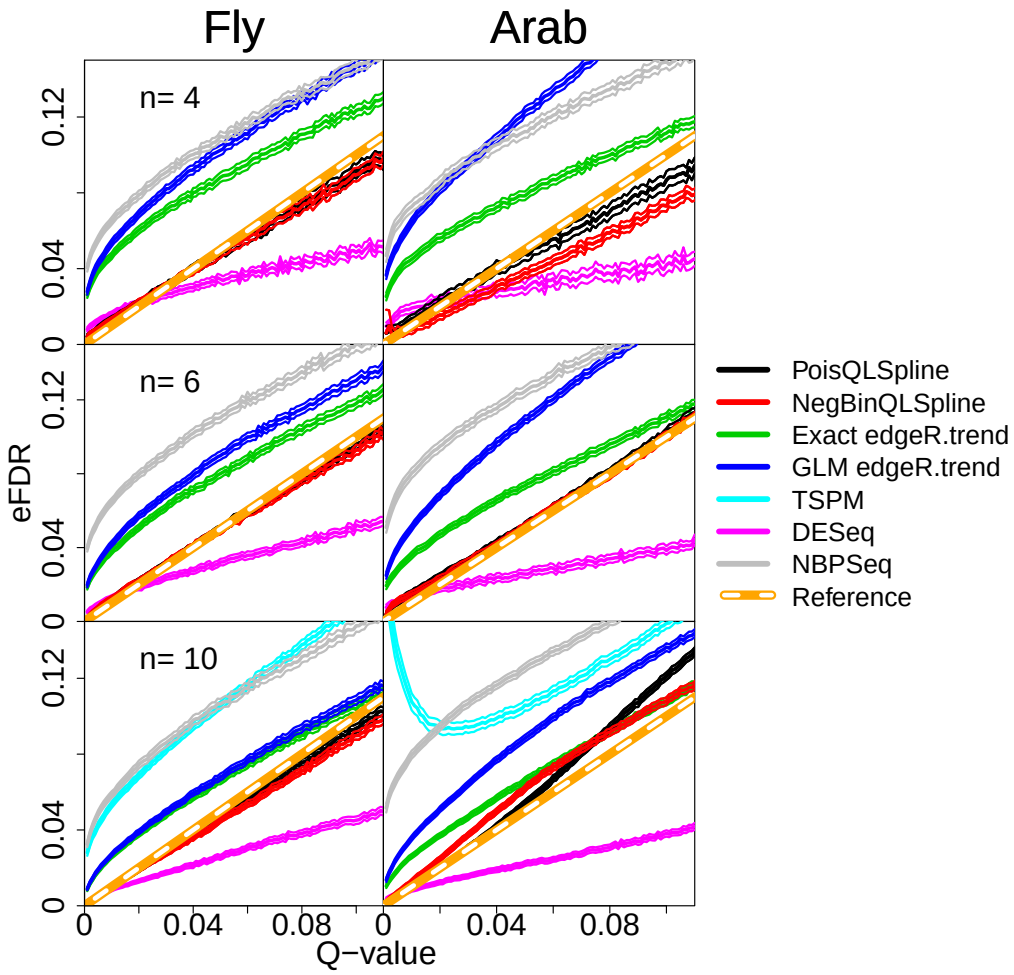

Figure 3.13 Curves relating average eFDR to q-values for negative binomial simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

10-fold NegBin and 10-fold perturbed simulations, respectively. To construct these plots, we rounded each q-value to the nearest 0.001 before plotting. When multiple genes produced identical rounded q-values for a given method, the eFDR of the gene with the largest original p-value was used to represent the set. (This technique facilitated averaging eFDRs across simulations and computing standard errors at each rounded q-value.) If a method was neither conservative nor liberal, its line should closely follow the dashed orange $y = x$ diagonal. Lines appearing substantially above or below the diagonal indicate the corresponding method was liberal or conservative, respectively.

The average eFDR curves for the TSPM, NBSeq, exact edgeR.trend, and GLM edgeR.trend meth-

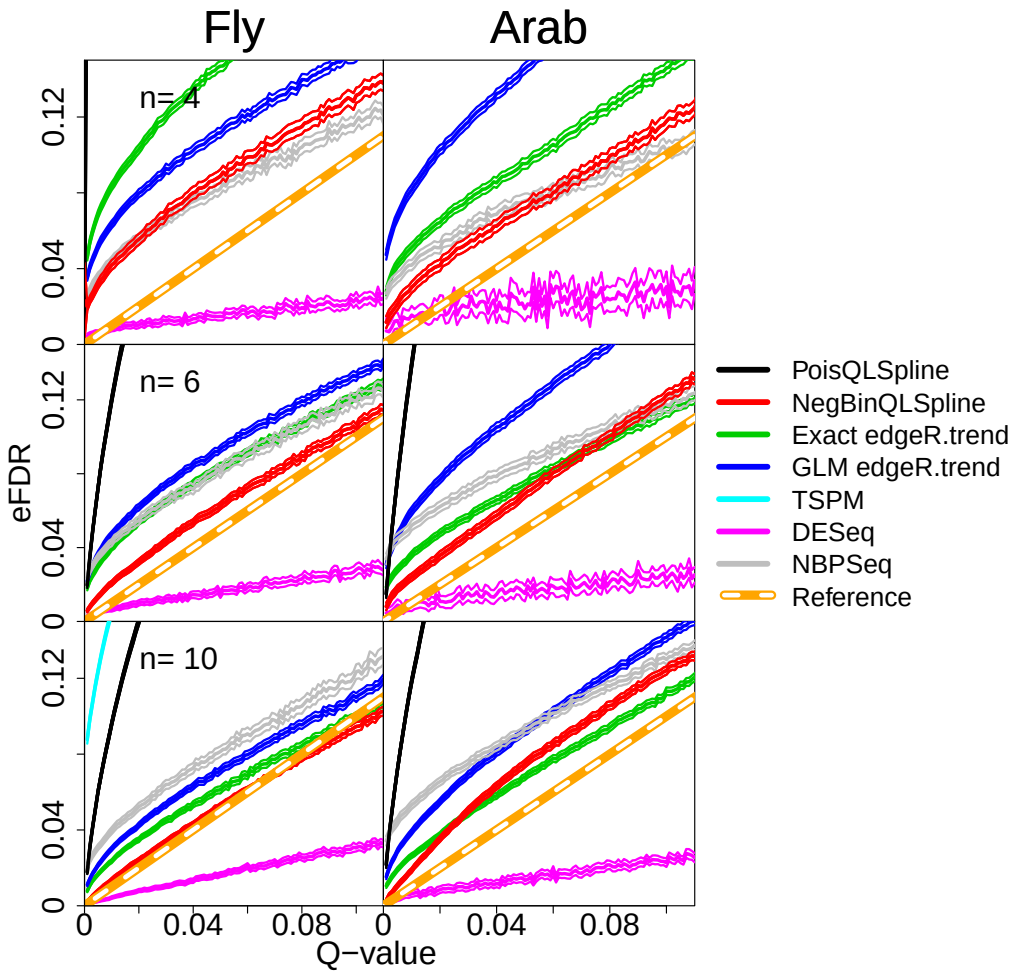

Figure 3.14 Curves relating average eFDR to q-values for 10-fold NegBin simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

ods are substantially above the dotted orange $y = x$ diagonal in every simulation scenario, indicating these methods produced liberal results for these data. DESeq was strongly conservative in these simulations. In simulations with moderate library size differences, the QLSpline methods produced accurate q-values. In simulations with 10-fold library size differences, PoisQLSpline produced severely liberal q-values. Q-values for NegBinQLSpline were moderately liberal and conservative in the 10-fold negative binomial and perturbed simulations, respectively.

The average eFDR with a corresponding q-value of 0.05 for each method are provided in Tables 3.3 through 3.8. Average eFDRs for DESeq and NegBinQLSpline were most often contained in (0.01,

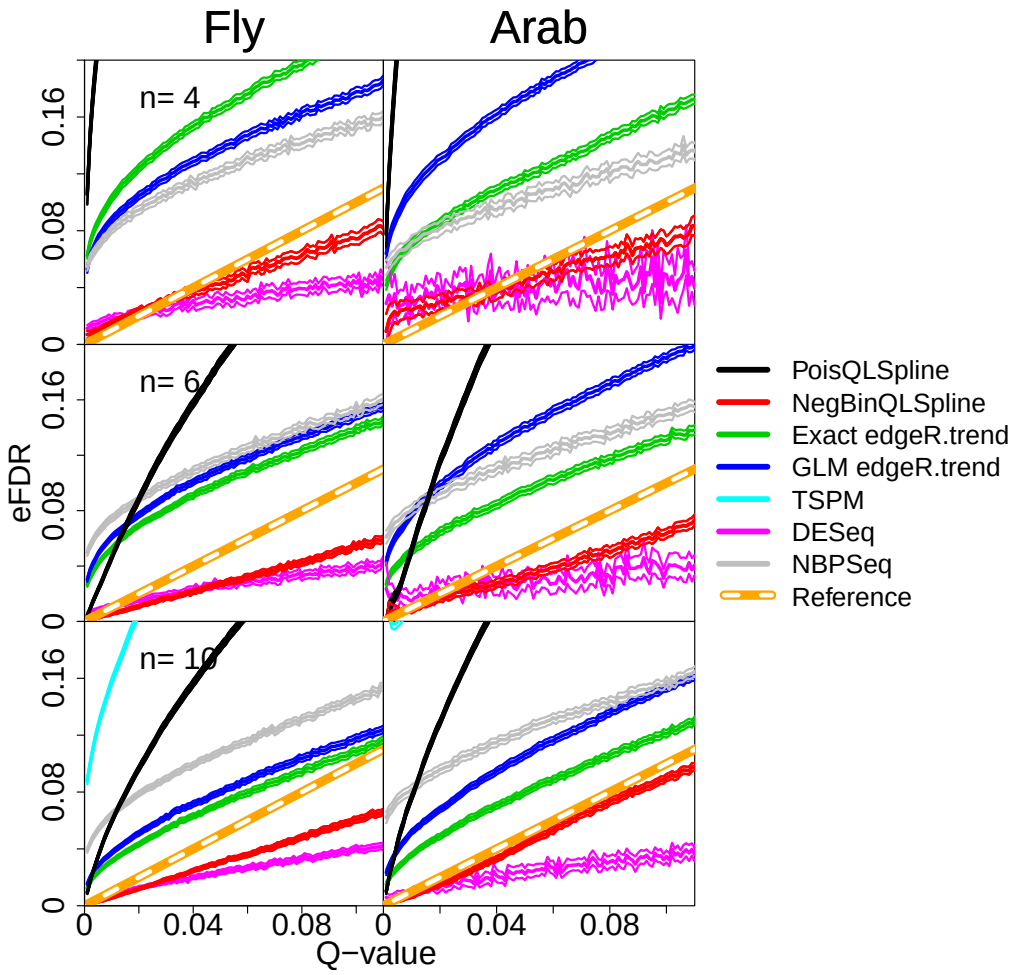

Figure 3.15 Curves relating average eFDR to q-values for 10-fold perturbed simulations based on fly embryo (left) and Arabidopsis (right) datasets with $n = 4$ (top), $n = 6$ (middle) and $n = 10$ (bottom).

0.03) and (0.03, 0.07), respectively. Average eFDRs for other methods were often substantially greater than 0.05. In the $n = 4$ 10-fold perturbed simulations based on the fly embryo dataset, for example, PoisQLSpline, TSPM, NBPSeq, and both edgeR methods all had average eFDRs greater than 0.12. To produce the most accurate q-values, we recommend using p-values obtained from NegBinQLSpline.

Interestingly, although many of the negative binomial modeling methods had liberal eFDRs compared to their q-values, they all underestimated the number of DE genes (1000) in every simulation scenario. DESeq was most conservative in this regard, with most of its estimates falling between 50 and 300. For DESeq, the number of genes with q-values less than 0.05 sometimes exceeded the estimated

Table 3.3 Summary of simulation results for negative binomial fly embryo simulations. Legend $\sim$ # DE Top 200: Number of truly DE genes contained in list of 200 most significant genes; $\text{eFDR}_{Q<.05}$: empirical FDR for list of all genes with q-values less than .05; $N_{Q<.05}$: Number of genes with q-values less than .05; $\hat{N}_{DE}$: Estimated number of DE genes; Max SE: Maximum standard error of averages.

Method	# DE Top 200	$\text{eFDR}_{Q<.05}$	$N_{Q<.05}$	$\hat{N}_{DE}$
		$n = 4$		
PoisQLSpline	196	0.0472	292.2	696
NegBinQLSpline	196.1	0.0472	290.5	663
Exact edgeR.trend	194.6$^{\star\circ}$	0.0869	379.7	476
GLM edgeR.trend	194.7$^{\star\circ}$	0.102	403.8	532
TSPM	148.8$^{\star\circ}$	0.44	498.1	806
DESeq	195.7$^{\star\circ}$	0.0352	238.6	242
NBPSeq	191.8$^{\star\circ}$	0.108	390	488
Max SE	0.45	0.00164	2.4	6.44
		$n = 6$		
PoisQLSpline	198.3	0.0495	367.5	710
NegBinQLSpline	198.3	0.0487	365.3	667
Exact edgeR.trend	197.3$^{\star\circ}$	0.0793	428.6	521
GLM edgeR.trend	197.3$^{\star\circ}$	0.0891	442.2	560
TSPM	161.2$^{\star\circ}$	0.243	387.9	802
DESeq	198.3	0.031	299.3	330
NBPSeq	193.3$^{\star\circ}$	0.11	441.4	539
Max SE	0.39	0.00162	1.7	5.64
		$n = 10$		
PoisQLSpline	199.8	0.0457	482.5	761
NegBinQLSpline	199.8	0.0442	476.9	721
Exact edgeR.trend	199.7$^{\star\circ}$	0.0656	521.4	594
GLM edgeR.trend	199.7$^{\star\circ}$	0.0668	524.1	614
TSPM	194$^{\star\circ}$	0.105	475.2	827
DESeq	199.8	0.0259	402.4	431
NBPSeq	196.7$^{\star\circ}$	0.107	538.6	620
Max SE	0.17	0.00132	1.7	5.99

$^{\circ}$ paired t-test comparing reported average to that of PoisQLSpline yielded two-sided p-value$<$0.01
* paired t-test comparing reported average to that of NegBinQLSpline yielded two-sided p-value$<$0.01

number of DE genes, which can be explained by the J-shape seen in its distribution of p-values from null simulated genes.

The impact of the suggested quasi-likelihood approaches can be illustrated by comparing results from NegBinQLSpline and GLM edgeR.trend, which are closely related. These methods share ω_k, the estimated negative binomial dispersion, for each gene and both use asymptotic tests for differential expression. Although both methods generally performed well, NegBinQLSpline has clear advantages. In each simulation scenario, the average number of truly DE genes contained in the list of 200 most significant genes was significantly greater for NegBinQLSpline than for GLM egdeR.trend. While q-values

Table 3.4 Summary of simulation results for negative binomial Arabidopsis simulations. See Table 3.3 for legend.

Method	# DE Top 200	$\text{eFDR}_{Q<.05}$	$N_{Q<.05}$	$\hat{N}_{DE}$
		$n = 4$		
PoisQLSpline	191.4*	0.0471	206.8	933
NegBinQLSpline	190.8°	0.0367	167.4	727
Exact edgeR.trend	190.4°	0.0784	304.4	512
GLM edgeR.trend	189.1$^{\star\circ}$	0.12	385.1	733
TSPM	126.4$^{\star\circ}$	0.624	516.4	1220
DESeq	189.5$^{\star\circ}$	0.0306	130.1	180
NBPSeq	183.1$^{\star\circ}$	0.112	280	508
Max SE	0.49	0.00158	4	11.9
		$n = 6$		
PoisQLSpline	194.9*	0.0499	319.5	977
NegBinQLSpline	194.5°	0.0503	299.3	760
Exact edgeR.trend	194$^{\star\circ}$	0.073	375.5	613
GLM edgeR.trend	192.9$^{\star\circ}$	0.108	442.9	757
TSPM	111.4$^{\star\circ}$	0.436	218	1180
DESeq	194.1°	0.0258	183.3	299
NBPSeq	184.2$^{\star\circ}$	0.123	355.6	611
Max SE	0.5	0.00257	2.1	8.7
		$n = 10$		
PoisQLSpline	198.5*	0.0529	486.4	1050
NegBinQLSpline	198.1°	0.0602	468.2	816
Exact edgeR.trend	197.8$^{\star\circ}$	0.0648	522.1	728
GLM edgeR.trend	197.4$^{\star\circ}$	0.0872	570.5	806
TSPM	181.3$^{\star\circ}$	0.104	315.5	1120
DESeq	197.9°	0.0215	302	451
NBPSeq	188.2$^{\star\circ}$	0.125	516.8	745
Max SE	0.3	0.00142	1.8	6.67

$^\circ$ paired t-test comparing reported average to that of PoisQLSpline yielded two-sided p-value$<$0.01
* paired t-test comparing reported average to that of NegBinQLSpline yielded two-sided p-value$<$0.01

for GLM edgeR.trend underestimated eFDRs in every simulation scenario, q-values for NegBinQLSpline were most often accurate or slightly conservative. The advantages of NegBinQLSpline are most clearly evident in the "10-fold perturbed" simulations, which demonstrates the robustness of the QL methods to model misspecification.

3.5 Discussion

The QL methods are only supported by asymptotic theory in special cases, as discussed in Section 3.2. However, this did not adversely affect their performance in our simulation study. Indeed, the NegBinQLSpline method provided better significance rankings and more accurate q-values than those for every alternative method in almost every simulation scenario. Other methods, like edgeR, DESeq

Table 3.5 Summary of simulation results for 10-fold NegBin simulations based on fly embryo data. See Table 3.3 for legend.

Method	# DE Top 200	eFDR$_{Q<.05}$	$N_{Q<.05}$	$\hat{N}_{DE}$
		$n = 4$		
PoisQLSpline	189.4*	0.545	1434.1	2360
NegBinQLSpline	193.6°	0.0879	345.3	552
Exact edgeR.trend	192.8$^{\star\circ}$	0.146	470.4	660
GLM edgeR.trend	193.2$^{\star\circ}$	0.112	403.1	524
TSPM	149.8$^{\star\circ}$	0.578	1290.2	2570
DESeq	192.4$^{\star\circ}$	0.0147	140.3	99
NBPSeq	190.8$^{\star\circ}$	0.0811	269.6	372
Max SE	0.47	0.00164	13.3	9.02
		$n = 6$		
PoisQLSpline	195.5*	0.32	769.7	1950
NegBinQLSpline	197.3°	0.0614	362.7	600
Exact edgeR.trend	197$^{\star\circ}$	0.0831	412.9	523
GLM edgeR.trend	196.6$^{\star\circ}$	0.0919	427.2	544
TSPM	142.7$^{\star\circ}$	0.418	799.9	1990
DESeq	195.4*	0.0172	179.5	168
NBPSeq	193.3$^{\star\circ}$	0.0825	310.3	421
Max SE	0.48	0.00154	3.1	9.3
		$n = 10$		
PoisQLSpline	199.4*	0.258	776.7	1760
NegBinQLSpline	199.7°	0.0515	464.9	662
Exact edgeR.trend	199.6°	0.061	488.1	571
GLM edgeR.trend	199.5*	0.0702	511.7	600
TSPM	184.8$^{\star\circ}$	0.299	774.3	1790
DESeq	199.4*	0.0158	281.7	299
NBPSeq	196.9$^{\star\circ}$	0.0806	398.6	515
Max SE	0.27	0.0014	2.5	10.6

$^\circ$ paired t-test comparing reported average to that of PoisQLSpline yielded two-sided p-value$<$0.01
* paired t-test comparing reported average to that of NegBinQLSpline yielded two-sided p-value$<$0.01

and NBPSeq, can test for differential expression between two treatments in a one-factor design using the exact test of Robinson and Smyth (2007). However, these methods also treat parameter estimates as true parameter values for their corresponding negative binomial distributions, which is also inaccurate and can produce an over-abundance of small p-values coming from EE genes. EdgeR, DESeq and NBPSeq methods use different dispersion estimates for each gene (for details, see McCarthy et al. (2012)), and regardless of estimation procedures, these estimates will have non-negligible uncertainties or biases for datasets with small values of $n - p$. While edgeR provides an option to assume a constant dispersion parameter common among all genes, this assumption has not been met in datasets we have examined.

When a relationship between estimated quasi-likelihood dispersions (as opposed to the dispersion in the variance function of the negative binomial distribution) and sample averages is present, the QL-

Table 3.6 Summary of simulation results for 10-fold NegBin simulations based on Arabidopsis data. See Table 3.3 for legend.

Method	# DE Top 200	eFDR$_{Q<.05}$	$N_{Q<.05}$	$\hat{N}_{DE}$
		$n = 4$		
PoisQLSpline	185.5*	0.599	1788.5	2840
NegBinQLSpline	188.1°	0.0715	232.5	712
Exact edgeR.trend	189.2$^{\star\circ}$	0.0954	321.2	658
GLM edgeR.trend	185.7*	0.144	392.6	775
TSPM	142.3$^{\star\circ}$	0.661	1481.8	2820
DESeq	182.1$^{\star\circ}$	0.0219	45.4	15.6
NBPSeq	184.9*	0.0753	196	402
Max SE	0.51	0.00376	10.9	9.32
		$n = 6$		
PoisQLSpline	186.9*	0.383	888.5	2350
NegBinQLSpline	192.4°	0.0679	306.1	771
Exact edgeR.trend	193$^{\star\circ}$	0.0732	350.2	628
GLM edgeR.trend	191$^{\star\circ}$	0.115	423.1	765
TSPM	81.7$^{\star\circ}$	0.499	814.6	2300
DESeq	186.8*	0.0165	62.3	104
NBPSeq	186.5*	0.085	260.1	500
Max SE	0.49	0.0023	3.6	10.4
		$n = 10$		
PoisQLSpline	193.7*	0.323	926.9	2170
NegBinQLSpline	197°	0.0737	456.7	831
Exact edgeR.trend	197.2°	0.0672	489.2	740
GLM edgeR.trend	196.6$^{\star\circ}$	0.0911	539.5	808
TSPM	163$^{\star\circ}$	0.345	782.4	2160
DESeq	193.9*	0.0147	119	255
NBPSeq	190.4$^{\star\circ}$	0.0915	400.8	649
Max SE	0.39	0.00137	3.2	9.54

$^\circ$ paired t-test comparing reported average to that of PoisQLSpline yielded two-sided p-value$<$0.01
* paired t-test comparing reported average to that of NegBinQLSpline yielded two-sided p-value$<$0.01

Spline method is generally preferable to the QLShrink method. The number of additional denominator degrees of freedom used in the QLShrink approach, $\hat{d}_0$, is estimated from the scatter of $\hat{\Phi}_k$ around a single constant for all k. The number of additional denominator degrees of freedom used in the QLSpline approach, $\hat{d}_0'$, is estimated from the scatter of $\hat{\Phi}_k$ around a spline fit to the (log-scale) relationship between $\hat{\Phi}_k$ and $\bar{y}_{\cdot\cdot k}$ for all k. When a relationship exists between sample means and estimated dispersions, the QLSpline method associates less random scatter with each $\hat{\Phi}_k$ than does the QLShrink method, which causes $\hat{d}_0'$ to be greater than $\hat{d}_0$. In the fly embryo dataset, for which $n - p = 2$, the PoisQLSpline and PoisQLShrink approaches produced estimates $\hat{d}_0' = 7.1$ and $\hat{d}_0 = 2.4$, respectively. Having more denominator degrees of freedom helps to increase the power of the QLSpline method over that of the QLShrink method. Separately, failing to account for the relationship with sample means

Table 3.7 Summary of simulation results for 10-fold perturbed simulations based on fly embryo data. See Table 3.3 for legend.

Method	# DE Top 200	$\text{eFDR}_{Q<.05}$	$N_{Q<.05}$	$\hat{N}_{DE}$
		$n = 4$		
PoisQLSpline	182.9*	0.449	1008.5	2050
NegBinQLSpline	190.6°	0.0424	190.2	356
Exact edgeR.trend	188.6$^{\star\circ}$	0.165	449.9	613
GLM edgeR.trend	187.7$^{\star\circ}$	0.137	379.9	494
TSPM	151.6$^{\star\circ}$	0.574	1256.1	2540
DESeq	185$^{\star\circ}$	0.0329	119	82.7
NBPSeq	181.9$^{\star\circ}$	0.12	256.4	358
Max SE	0.44	0.00208	6.7	9.82
		$n = 6$		
PoisQLSpline	192.3*	0.185	462.8	1460
NegBinQLSpline	196.1°	0.0267	227.7	390
Exact edgeR.trend	194.4$^{\star\circ}$	0.0988	390.5	497
GLM edgeR.trend	193.8$^{\star\circ}$	0.109	400.2	507
TSPM	147.7$^{\star\circ}$	0.417	792.8	1970
DESeq	190.2$^{\star\circ}$	0.0261	148.8	145
NBPSeq	186$^{\star\circ}$	0.116	294.8	406
Max SE	0.44	0.00154	2.9	9.15
		$n = 10$		
PoisQLSpline	198.6*	0.18	603.5	1410
NegBinQLSpline	199.6°	0.0302	376.1	499
Exact edgeR.trend	199$^{\star\circ}$	0.071	471.4	546
GLM edgeR.trend	198.9$^{\star\circ}$	0.0791	488.1	561
TSPM	185.9$^{\star\circ}$	0.302	767.6	1770
DESeq	197.8$^{\star\circ}$	0.0235	247.9	273
NBPSeq	192.7$^{\star\circ}$	0.107	388.5	491
Max SE	0.25	0.00156	2.8	10.5

$^{\circ}$ paired t-test comparing reported average to that of PoisQLSpline yielded two-sided p-value$<$0.01
* paired t-test comparing reported average to that of NegBinQLSpline yielded two-sided p-value$<$0.01

when shrinking estimated dispersions can induce bias. For example, if there is an increasing relationship between average counts and dispersion, then shrinking each estimated dispersion toward a single central value will systematically underestimate (overestimate) dispersions for genes with large (small) average counts.

When implementing the QLSpline methods, we suggest restricting the set of analyzed genes to include only those for which the average count across all samples is at least one and for which at least two samples have positive counts. This general guideline has been appropriate for both real and simulated data originating from single factor experimental designs with a moderate number of levels. Experimental designs with more than one factor, like the analysis of the Arabidopsis dataset that included block effects, may require more selective criteria when estimating dispersions. Users can examine a scatter-

Table 3.8 Summary of simulation results for 10-fold perturbed simulations based on Arabidopsis data. See Table 3.3 for legend.

Method	# DE Top 200	$\text{eFDR}_{Q<.05}$	$N_{Q<.05}$	$\hat{N}_{DE}$
		$n = 4$		
PoisQLSpline	174.5*	0.526	1245.2	2550
NegBinQLSpline	181°	0.042	86.5	387
Exact edgeR.trend	181.3°	0.116	255.5	554
GLM edgeR.trend	176.9$^{\star\circ}$	0.175	326.2	662
TSPM	136.8$^{\star\circ}$	0.67	1415.6	2750
DESeq	169.3$^{\star\circ}$	0.0421	35.5	0.653
NBPSeq	175.2*	0.103	154.8	340
Max SE	0.48	0.00517	8.3	8.77
		$n = 6$		
PoisQLSpline	176.6*	0.249	429.2	1940
NegBinQLSpline	186.9°	0.0315	118	476
Exact edgeR.trend	186.2$^{\star\circ}$	0.0941	262.9	536
GLM edgeR.trend	183.2$^{\star\circ}$	0.14	328.7	661
TSPM	85.4$^{\star\circ}$	0.515	761	2260
DESeq	174.6$^{\star\circ}$	0.0299	44.3	50.5
NBPSeq	176.5*	0.118	199.5	422
Max SE	0.5	0.00333	3.3	10.6
		$n = 10$		
PoisQLSpline	186.5*	0.242	557.5	1870
NegBinQLSpline	193°	0.0423	223.3	592
Exact edgeR.trend	192.8°	0.0786	355.6	637
GLM edgeR.trend	191.5$^{\star\circ}$	0.105	400	693
TSPM	160$^{\star\circ}$	0.359	677.2	2100
DESeq	184.2$^{\star\circ}$	0.0246	69	166
NBPSeq	181.1$^{\star\circ}$	0.123	296.4	559
Max SE	0.35	0.00245	3	8.57

$^\circ$ paired t-test comparing reported average to that of PoisQLSpline yielded two-sided p-value$<$0.01
* paired t-test comparing reported average to that of NegBinQLSpline yielded two-sided p-value$<$0.01

plot of $\hat{\Phi}_k$ versus $\bar{y}_{\cdot k}$ and the distribution of p-values to assess if their inclusion criteria is restrictive enough. If an isolated cluster of points appears at small values of $\hat{\Phi}_k$ versus $\bar{y}_{\cdot k}$ relative to the main body of points, the required number of samples with nonzero counts may need to be modified. If there is a sharp peak around one in the distribution of p-values for the QLSpline methods, the required average count may need to be increased.

The best significance rankings in each scenario came from the QLSpline method applied to either a quasi-Poisson or quasi-negative binomial model, and p-values from one of the QLSpline methods also produced q-values that most closely followed empirical FDRs. For moderate differences among library sizes, the QLSpline methods produced similar results. For datasets with large differences between library sizes, NegBinQLSpline clearly outperformed PoisQLSpline. The authors therefore recommend

NegBinQLSpline among the methods included in QuasiSeq. Intuitively pleasing, the QL (QLShrink and QLSpline) methods quantify the effect of parameter constraints in terms of residual degrees of freedom in an approach analogous to ANOVA (with shrunken variance estimates) and are robust to model misspecification. The implementation of the suggested methods via the QuasiSeq package is fast, simple and flexible enough to handle all models that can be analyzed by an ordinary GLM.

3.6 QuasiSeq Package Demonstration on Arabidopsis Dataset

The authors have developed an R (R Development Core Team, 2011) package called QuasiSeq, available from the CRAN website, used to implement the suggested methods of this article. Code used to analyze the Arabidopsis dataset described in Section 3.3 with the quasi-Poisson model and some selected results are shown below.

3.6.1 Analysis of Arabidopsis data without block effects

```
> ##Load QuasiSeq
> library(QuasiSeq);
> ##Load data
> library(NBPSeq); data(arab); counts=arab;
> ##Change duplicate gene ID
> row.names(counts)[row.names(counts)=="AT4G32850"][2]="AT4G32850.2"
>
>  ## Only use genes with at least 7 total counts
>  ## and at least 2 samples with positive counts
> counts<-as.matrix(counts[rowSums(counts>0)>1&
+ rowSums(counts)>ncol(counts),])
>
>  ## View first 6 rows of data
>  head(counts)
          mock1 mock2 mock3 hrcc1 hrcc2 hrcc3
AT1G01010    35    77    40    46    64    60
AT1G01020    43    45    32    43    39    49
AT1G01030    16    24    26    27    35    20
```

```
AT1G01040    72    43    64    66    25    90
AT1G01050    49    78    90    67    45    60
AT1G01060     0    15     2     0    21     8
>
>  ## Define models under alternative and null hypotheses
>  design.list<-vector("list", 2)
>    # Model under alternative hypothesis (DE gene)
>  design.list[[1]]<-rep(1:2, each=3)
>    # Model under null hypothesis (EE gene)
>  design.list[[2]]<-rep(1, ncol(counts))
>
>  ## estimate library size factors
>  size<-apply(counts, 2, quantile, .75)
>
>  ## load QLSpline package and fit data
>  #library(QLSpline)
>  res<-QL.fit(counts, design.list, log.offset=log(size),Model="Poisson")
>  results<-QL.results(QL.fit=res)
[1] "Spline scaling factor: 1.13041121749462"
>
>  ## How many genes have q-values less than 0.05?
>  apply(results$Q.values<.05, 2, sum)
    Poisson QL Poisson QLShrink d0=3.3 Poisson QLSpline d0=7.1
             0                     386                     203
>
>  ## What is the estimated number of DE genes?
>  round(nrow(counts)-results$m0)
    Poisson QL Poisson QLShrink d0=3.3 Poisson QLSpline d0=7.1
          2045                    2103                    2242
```

3.6.2 Analysis of Arabidopsis data with block effects

```
> ## Only use genes with at least 3 total samples
```

```
> ## (and at least 1 sample from both treatments)
> ## with positive counts and at least 7 total counts
> counts<-as.matrix(counts[rowSums(counts>0)>2&
+ rowSums(counts[,1:3]>0)>0&rowSums(counts[,4:6]>0)>0&
+ rowSums(counts)>ncol(counts),])
>
>  ## Define block and treatment levels
> block<-rep(1:3,2)
> trt<-rep(1:2,each=3)
>
> ## Define model designs
> design.list<-vector("list",3)
>
> ## Full model includes both
> ## block and treatment effects
> design.list[[1]]<-model.matrix(~as.factor(block)
+ +as.factor(trt))
>
> ## Test for block effects using
> ## design with only treatment effects
> design.list[[2]]<-trt
>
> ## Test for treatment effects using
> ## design with only block effects
> design.list[[3]]<-block
>
>  res<-QL.fit(counts,design.list,log.offset=log(size),Model="Poisson")
>  results2<-QL.results(QL.fit=res)
[1] "Spline scaling factor: 1.66845267016672"
>
> ## How many genes have q-values less than 0.05
> ## for the test of block effects?
```

```
>  apply(results2$Q.values[[1]]<.05, 2, sum)
     Poisson QL  Poisson QLShrink d0=5.1 Poisson QLSpline d0=15.6
               0                     4354                     6133
>  ## What is the estimated number of genes with block effects?
>  round(nrow(counts)-results2$m0[1,])
       Poisson QL Poisson QLShrink d0=5.1 Poisson QLSpline d0=15.6
LRT1 2      11144                   12044                    12094
>
> ## How many genes have q-values less than 0.05
> ## for the test of treatment effects?
>  apply(results2$Q.values[[2]]<.05, 2, sum)
     Poisson QL  Poisson QLShrink d0=5.1 Poisson QLSpline d0=15.6
               0                     2326                     2802
>  ## What is the estimated number of DE genes?
>  round(nrow(counts)-results2$m0[2,])
       Poisson QL Poisson QLShrink d0=5.1 Poisson QLSpline d0=15.6
LRT1 3       5964                    6588                     6911
```

CHAPTER 4. INCORPORATING RNA-SEQ MULTIREADS WHEN TESTING FOR DIFFERENTIAL EXPRESSION

4.1 Introduction

The previous chapter discussed the analysis of RNA-seq reads that map to a single transcript. For most RNA-seq experiments, there are many reads that align with multiple transcripts. A read that aligns with more than one transcript is called a multiread. A standard approach for handling multireads is to simply omit multireads from transcript quantification. This approach reduces power by discarding valuable information. This chapter extends the quasi-Poisson methods of Chapter 3 to incorporate multireads when testing for differential expression.

The proportion of reads that align with multiple genes varies widely depending on the transcriptome and read length. A recent article on this topic (Li et al., 2010a) analyzed RNA-seq data sets collected from mice and maize in which the proportion of all reads that mapped to multiple genes were 17% and 52%, respectively. In some experiments, researchers are interested in examining the expression levels of various isoforms of a single gene. When each isoform is considered to be a separate transcript, the proportion of multireads in the data can increase dramatically. For a given transcriptome, increasing read length will decrease the proportion of multireads, but will also require more extensive sequencing to provide the same overall number of reads. Li et al. (2010a) suggests, for fixed sequencing throughput, that read lengths of 20-25 bases are optimal for gene-level expression estimation from the mouse and maize RNA-Seq data. For most transcriptomes, using read lengths of 20-25 bases would produce a substantial number of multireads.

The PoisQL methods described in Chapter 3 provides a convenient framework for working with multireads. One strength of the QL methods is that the numerator and denominator of their F statistics are derived from fitting ordinary generalized linear models, which are both tractable and flexible. Section 4.2 describes the how to model multireads by extending the GLMs used in Chapter 3. Section 4.3 presents a series of simulation studies that demonstrate using our suggested method for incorporating multireads provides improved significance rankings when compared to the PoisQL and PoisQLSpline methods that discard multireads. Section 4.4 provides some discussion regarding future work and generalizations to other models besides quasi-Poisson. Because this chapter focuses on the quasi-Poisson model, we henceforth drop the "Pois" prefix so the "PoisQL" and "PoisQLSpline"

methods from Chapter 3 are now called the "QL" and "QLSpline" methods, respectively.

4.2 Method Description

This paper describes how to model multireads that map to two transcripts, which represent the majority of multireads. However, these approaches can also handle multireads that map to three or more transcripts. Consider fitting a quasi-Poisson GLM for each transcript. Let U_{ik} represent the count of uniquely mapping reads, which we refer to as "unireads," attributed to transcript k from the ith sample ($i = 1, \ldots, I$). Let $M_{ikk'}$ represent the count of multireads shared between transcripts k and k' from the ith sample. Let Y_{ik} represent the total number of reads (including unireads and some unknown portion of the multireads) originating from transcript k in sample i. Note that Y_{ik} is not observed, because we do not know which multireads involving transcript k actually originated from transcript k. Let c_i represent a normalization factor for the overall number of reads from the ith sample (e.g., we set c_i as the 0.75 quantile of unireads counts from the ith sample as recommended by Bullard et al. (2010)). Let $\mu_{ik} = \lambda_{ik} c_i$ represent the cumulative expression level of all reads from transcript k in sample i. We assume $Y_{ik} \sim \text{quasi-Poisson}(\mu_{ik}, \Phi_k)$, allowing each transcript to have its own unknown dispersion parameter, Φ_k, such that $\text{Var}(Y_{ik}) = \mu_{ik}\Phi_k$.

For each transcript, we fit the model $\log(\lambda_{ik}) = \mathbf{X}_i \beta_k$, where $\mathbf{X}_i$ is the row of the experiment's design matrix pertaining to sample i and β_k is a vector of model parameters for transcript k. As an example, suppose a completely randomized experiment was conducted to examine the effect of a categorical treatment factor. The model for this example can be written as $\log(\lambda_{ik}) = \beta_{0k} + \beta_{\tau(i)k}$, where $\tau(i)$ denotes the index of the treatment applied to the ith sample. In this expression, $\exp(\beta_{0k})$ represents a baseline level of combined expression of all reads from transcript k when $c_i = 1$, and $\exp(\beta_{\tau(i)k})$ represents the multiplicative effect of the $\tau(i)$th treatment on the baseline level of expression of all reads from transcript k. Depending on the experimental design, other covariates, such as indicators for the levels of blocking factors, may also be added to the model.

Count data for both unireads and multireads can provide information for estimating β_k. Modeling multireads is more involved than modeling unireads, as multiread counts provide the sum of two (or more) latent variables, where the individual latent variables themselves are directly reltaed to β_k. We develop models to accommodate multireads starting with the assumption that reads are uniformly distributed within each transcript. That is, we assume that within a transcript each possible read of a given length is equally likely to be observed. This assumption allows us to attribute portions of multireads to their possible origins according to transcript lengths, which are well known. In the following description "length" is used to denote the number of bases contained in a transcript or read sequence.

Suppose a genome consists of K distinct transcripts and that L_k denotes the length of transcript k. Let R denote

the fixed read length of the considered RNA-seq data set. Assume that within each transcript each sequence of length R appears at most once (i.e. each sequence of length R within a transcript is distinct from all other sequences of length R within the same transcript). Then the set of all possible reads fully contained within transcript k has $TL_k = L_k - R + 1$ members. We refer to TL_k as the total "effective length" of transcript k. The total effective length of transcript k is equal to the sum of the effective length of regions of transcript k that would produce unireads and the effective lengths of regions of transcript k that would produce multireads with other transcripts.

Let $ML_{kk'}$ denote the effective length of the regions producing multireads shared between transcripts k and k' (i.e. the total number sequences of length R that appear in both transcripts k and k'). Note that $ML_{kk'} = ML_{k'k}$. As an example, suppose a sequence of length $L_{12} > R$ appears in transcripts 1 and 2, and that no other sequence longer than $R-1$ bases appears in both transcripts. In this case, there are $ML_{12} = L_{12} - R + 1$ possible reads fully contained within the sequence that appears in both transcripts. Let G_k be the set of transcript indices k' such that $ML_{kk'} > 0$. Then the effective length of the regions of transcript k that would produce a uniread is given by $UL_k = TL_k - \sum_{k' \in G_k} ML_{kk'}$.

Although the effective lengths of overlaps between transcripts can be determined with extensive knowledge of transcript sequences, values for $ML_{kk'}$ terms are not often readily available. One option for handling the $ML_{kk'}$ terms, presented below, is to consider them as model parameters to be estimated during the optimization of quasi-likelihoods. A second option, presented later in this section, is to first estimate the $ML_{kk'}$ terms using the EM algorithm and to consider the resulting values as fixed and known before estimating other model parameters by quasi-likelihood optimization.

Assuming each possible read in a transcript is equally likely to occur, the average proportion of unireads from the set of all reads generated from transcript k is given by UL_k/TL_k. Similarly, the average proportion of all reads produced by transcript k that are multireads with transcript k' is given by $ML_{kk'}/TL_k$. We therefore assume

$$U_{ik}|UL_k, TL_k, \mu_{ik} \sim \text{quasi-Poisson}\left(\frac{UL_k}{TL_k}\mu_{ik}\right)$$

and

$$M_{ikk'}|ML_{kk'}, TL_k, TL_{k'}, \mu_{ik}, \mu_{ik'} \sim \text{quasi-Poisson}\left(\frac{ML_{kk'}}{TL_k}\mu_{ik} + \frac{ML_{kk'}}{TL_{k'}}\mu_{ik'}\right)$$

. While we assume that observed counts are independent from one transcript to another, the quasi-likelihoods of all transcripts within a "multiread network" are interdependent.

The grouping structure of multireads can be thought of as a social network and can be used to partition the set of all transcripts. We say transcript k is "friends" with each transcript that it shares at least one multiread with. To establish a closed network containing transcript k, one begins with the set containing transcript k and each of its friends. All transcripts outside the network that have a friend in the network are added, and the process repeats

until no transcript inside the network has any friends outside the network. Let S_k denote the "multiread network" containing transcript k. The log quasi-likelihood (q-likelihood) for set S_{k*} when treating $ML_{kk'}$ terms as unknown parameters is given by

$$\ell_{S_{k*}}\left(\{\beta_k : k \in S_{k*}\}, \{ML_{kk'} : (k,k') \in S^2_{k*}\} | \{TL_k : k \in S_{k*}\}, \mathbf{C}\right) = \tag{4.1}$$
$$\sum_i \left\{ \sum_{k \in S_{k*}} \left[u_{ik} \log\left(\frac{UL_k}{TL_k}\mu_{ik}\right) - \frac{UL_k}{TL_k}\mu_{ik} - \log(u_{ik}!)\right] \right.$$
$$\left. + \sum_{(k,k') \in S^2_{k*}} \left[m_{ikk'} \log(\gamma_{ikk'}) - \gamma_{ikk'} - \log(m_{ikk'}!)\right] \right\}$$

where $\mathbf{C} = (c_1, c_2, \ldots, c_I)'$, $\mu_{ik} = \exp(X_i'\beta_k)c_i$, $UL_k = TL_k - \sum_{k' \in G_k} ML_{kk'}$, and $\gamma_{ikk'} = \frac{ML_{kk'}}{TL_k}\mu_{ik} + \frac{ML_{kk'}}{TL_{k'}}\mu_{ik'}$.

As in Chapter 3, a test for differential expression involves optimizing the log q-likelihood for a transcript (and its multiread network) under a full and reduced model. The optimized q-likelihood from the full model can be shared among all transcripts within a multiread network. However, a reduced model that places constraints only on β parameters for transcript k, leaving the β parameters for all other transcripts in the multiread network unconstrained, must be fit for each transcript k. The naïve test statistic is given as twice the difference between the optimized log quasi-likelihoods from the full and reduced models. To adjust for overdispersion, we propose estimating the dispersion for each transcript, Φ_k, based on its unireads following the QLSpline method proposed in Section 2 of Chapter 3. We then suggest comparing the ratio $LRT_k/(q\tilde{\Phi}_k^{(spline)})$ to an F-distribution with q and $\hat{d}_0' + n - p$ degrees of freedom.

Some genomes may produce large multiread networks, which may create problems by requiring the simultaneous estimation of many β and ML parameters. One can address this issue by using a restrictive subset of a multiread network (e.g. include only the transcript being tested, its friends, and its friends' friends). The smallest usable network is simply the transcript being tested, which we call the "target" transcript. This is the network used by all popular methods for detecting differential expression of which we are aware. The smallest appropriate network that includes multireads would consist of the target transcript and its friends. Modeling this reduced network involves specifying a q-likelihood function for unireads of the target transcript and its friends and multireads involving the target transcript. When using a reduced network, the effective length of the target transcript's friends for whom some multireads have been excluded should be adjusted to reflect the excluded portion of their counts.

One can make such adjustments using estimates of the effective lengths of regions for each possible multiread pair involving a transcript that is still modeled and a transcript that is excluded. Such estimates can be obtained from count data using the EM algorithm in the following manner. Let $u_{\cdot k}$ and $m_{\cdot kk'}$ denote the sum across samples of unireads from transcript k and multireads shared between transcripts k and k', respectively. Let $v_{\cdot k(k')}$ denote the

unobserved total number of reads originating from transcript k that map to transcripts k and k', such that $m_{\cdot kk'} = \nu_{\cdot k(k')} + \nu_{\cdot k'(k)}$. Let $\lambda_{\cdot k}$ represent the combined expression level across all samples for all reads originating from transcript k. The complete data is given by $\{u_{\cdot k}\}_{k=1}^{K}$ and $\{\nu_{\cdot k(k')}\}_{k\neq k'}$, where $u_{\cdot k} \sim$quasi-Poisson$\left(\frac{UL_k}{TL_k}\lambda_{\cdot k}\right)$ where $ML_k = \sum_{k'\neq k} ML_{kk'}$ and $\nu_{\cdot k(k')} \sim$quasi-Poisson$\left(\frac{ML_{kk'}}{TL_k}\lambda_{\cdot k}\right)$. Let $\Delta = (\lambda_{\cdot 1},\ldots,\lambda_{\cdot K}, ML_{12}, ML_{13},\ldots,ML_{(K-1)K})$ be the vector of unknown model parameters. The log q-likelihood for the complete data is then given by

$$\ell(\Delta|\{u_{\cdot k}\}_{k=1}^{K},\{\nu_{\cdot k(k')}\}_{k\neq k'}) = \sum_{k}\left\{u_{\cdot k}\log\left(\frac{TL_k - ML_k}{TL_k}\lambda_{\cdot k}\right) - \lambda_{\cdot k} + \sum_{k'\neq k}\left[\nu_{\cdot k(k')}\log\left(\frac{ML_{kk'}}{TL_k}\lambda_{\cdot k}\right)\right]\right\}. \tag{4.2}$$

The conditional distribution and expectation of the latent variables $\nu_{\cdot k(k')}$ are given by

$$\nu_{\cdot k(k')}|m_{\cdot kk'},\Delta \sim \text{binomial}\left(m_{\cdot kk'}, \frac{\lambda_{\cdot k}/TL_k}{\lambda_{\cdot k}/TL_k + \lambda_{\cdot k'}/TL_{k'}}\right) \tag{4.3}$$

and

$$\nu^{*}_{\cdot k(k')} = \mathrm{E}(\nu_{\cdot k(k')}|m_{\cdot kk'},\Delta) = m_{\cdot kk'}\frac{\lambda_{\cdot k}/TL_k}{\lambda_{\cdot k}/TL_k + \lambda_{\cdot k'}/TL_{k'}}. \tag{4.4}$$

Thus, the expectation (E) step of the EM algorithm during the itth iteration consists of computing the conditional expectation of the log-likelihood for the complete data, given by

$$\begin{aligned} Q(\Delta) &= \mathrm{E}_{\Delta}(\ell(\Delta|\{u_{\cdot k}\}_{k=1}^{K},\{\nu_{\cdot k(k')}\}_{k\neq k'})|\{u_{\cdot k}\}_{k=1}^{K},\{m_{\cdot kk'}\}_{k\neq k'}) \qquad (4.5)\\ &= \sum_{k}\left\{u_{\cdot k}\left(\frac{TL_k - ML_k}{TL_k}\lambda_{\cdot k}\right) - \lambda_{\cdot k} + \sum_{k'\neq k}\left[\nu^{*(it)}_{\cdot k(k')}\left(\frac{ML_{kk'}}{TL_k}\lambda_{\cdot k}\right)\right]\right\}, \end{aligned}$$

where

$$\nu^{*(it)}_{\cdot k(k')} = \mathrm{E}(\nu_{\cdot k(k')}|m_{\cdot kk'},\Delta^{(it)}) = m_{\cdot kk'}\frac{\lambda^{(it)}_{\cdot k}/TL_k}{\lambda^{(it)}_{\cdot k}/TL_k + \lambda^{(it)}_{\cdot k'}/TL_{k'}}. \tag{4.6}$$

The maximization (M) step of the EM algorithm involves finding roots to the partial derivatives,

$$\frac{\partial Q(\Delta)}{\partial \lambda_{\cdot k}} = -1 + \frac{u_{\cdot k} + \sum_{k'\neq k}\nu^{*(it)}_{\cdot k(k')}}{\lambda_{\cdot k}} \tag{4.7}$$

and

$$\frac{\partial Q(\Delta)}{\partial ML_{kk'}} = \frac{m_{\cdot kk'}}{ML_{kk'}} - \frac{u_{\cdot k}}{TL_k - ML_k} - \frac{u_{\cdot k'}}{TL_{k'} - ML_{k'}}. \tag{4.8}$$

The root of (4.7) is given by

$$\hat{\lambda}^{(it+1)}_{\cdot k} = u_{\cdot k} + \sum_{k'\neq k}\nu^{*(it)}_{\cdot k(k')} = u_{\cdot k} + \sum_{k'\neq k}\left(m_{\cdot kk'}\frac{\lambda^{(it)}_{\cdot k}/TL_k}{\lambda^{(it)}_{\cdot k}/TL_k + \lambda^{(it)}_{\cdot k'}/TL_{k'}}\right), \tag{4.9}$$

which does not involve $ML_{kk'}$. Thus, one may ignore $ML_{kk'}$ until after the λ_k parameters have converged.

Let $\hat{\lambda}_{\cdot k}$ denote transcript k's solution in the equations

$$\hat{\lambda}_{\cdot k} = u_{\cdot k} + \frac{\hat{\lambda}_{\cdot k}}{TL_k}\sum_{k'\neq k}\left(\frac{m_{\cdot kk'}}{\hat{\lambda}_{\cdot k}/TL_k + \hat{\lambda}_{\cdot k'}/TL_{k'}}\right), \tag{4.10}$$

obtained by the iterative process defined in (4.9).

Let $A_{kk'} = \hat{\lambda}_{\cdot k}/TL_k + \hat{\lambda}_{\cdot k'}/TL_{k'}$. Our proposed estimator of $ML_{kk'}$ is $\widehat{ML}_{kk'} = m_{\cdot kk'}/A_{kk'}$. We now show that $\widehat{ML}_{kk'}$ is a root of (4.8).

Setting (4.8) to zero implies $\frac{m_{\cdot kk'}}{ML_{kk'}} = \frac{u_{\cdot k}}{TL_k - ML_k} + \frac{u_{\cdot k'}}{TL_{k'} - ML_{k'}}$. Therefore to prove that $\widehat{ML}_{kk'}$ provides a root of (4.8), we must show

$$A_{kk'} = \frac{u_{\cdot k}}{TL_k - \widehat{ML}_k} + \frac{u_{\cdot k'}}{TL_{k'} - \widehat{ML}_{k'}}, \tag{4.11}$$

where $\widehat{ML}_k = \sum_{k' \neq k} \widehat{ML}_{kk'}$.

Solving (4.10) for u_k and substituting into the right hand side of (4.11) for u_k and $u_{k'}$ yields

$$\begin{aligned}
\frac{u_{\cdot k}}{TL_k - \widehat{ML}_k} + \frac{u_{\cdot k'}}{TL_{k'} - \widehat{ML}_{k'}} &= \frac{\hat{\lambda}_{\cdot k} - \frac{\hat{\lambda}_{\cdot k}}{TL_k} \sum_{k^* \neq k} \frac{m_{kk^*}}{A_{kk^*}}}{TL_k - \widehat{ML}_k} + \frac{\hat{\lambda}_{\cdot k'} - \frac{\hat{\lambda}_{\cdot k'}}{TL_{k'}} \sum_{k^* \neq k'} \frac{m_{k'k^*}}{A_{k'k^*}}}{TL_{k'} - \widehat{ML}_{k'}} \\
&= \frac{\frac{\hat{\lambda}_{\cdot k}}{TL_k} \left(TL_k - \sum_{k^* \neq k} \frac{m_{kk^*}}{A_{kk^*}} \right)}{TL_k - \widehat{ML}_k} + \frac{\frac{\hat{\lambda}_{\cdot k'}}{TL_{k'}} \left(TL_{k'} - \sum_{k^* \neq k'} \frac{m_{k'k^*}}{A_{k'k^*}} \right)}{TL_{k'} - \widehat{ML}_{k'}} \\
&= \frac{\frac{\hat{\lambda}_{\cdot k}}{TL_k}(TL_k - \widehat{ML}_k)}{TL_k - \widehat{ML}_k} + \frac{\frac{\hat{\lambda}_{\cdot k'}}{TL_{k'}}(TL_{k'} - \widehat{ML}_{k'})}{TL_{k'} - \widehat{ML}_{k'}} \\
&= \frac{\hat{\lambda}_{\cdot k}}{TL_k} + \frac{\hat{\lambda}_{\cdot k'}}{TL_{k'}}
\end{aligned} \tag{4.12}$$

EOP

The resulting implementation of the EM algorithm can be summarized as follows.

- Step 0: Initialize $\lambda_{\cdot k}$ parameters (e.g. $\hat{\lambda}_{\cdot k}^{(0)} = u_{\cdot k} + 1, \forall k \in S_{k*}$).
- Step 1: $\hat{\lambda}_{\cdot k}^{(it+1)} = u_{\cdot k} + \sum_{k' \in \tau_k} \nu_{\cdot k(k')}^{*(it)}$, where $\nu_{\cdot k(k')}^{*(it)} = \frac{\hat{\lambda}_{\cdot k}^{(it)}/TL_k}{\hat{\lambda}_{\cdot k}^{(it)}/TL_k + \hat{\lambda}_{\cdot k'}^{(it)}/TL_{k'}} m_{\cdot kk'}$ is the portion of $m_{\cdot kk'}$ estimated to have come from transcript k. Repeat Step 1 until convergence.

Following convergence, the effective length of the overlapping regions between transcripts k and k' can be estimated by $\widehat{ML}_{kk'} = TL_k \nu_{\cdot k(k')}^{*(IT)} / \hat{\lambda}_{\cdot k}^{(IT)} = TL_{k'} \nu_{\cdot k'(k)}^{*(IT)} / \hat{\lambda}_{\cdot k'}^{(IT)}$, where IT is the total number of iterations of Step 1 required to reach convergence. The corresponding estimate of the effective length for the uniquely mapping regions of transcript k is given by $\widehat{UL}_k = TL_k u_{\cdot k} / \hat{\lambda}_{\cdot k}^{(IT)} = TL_k - \sum_{k' \in G_k} \widehat{ML}_{kk'}$. In the q-likelihood functions used to conduct a hypothesis test on transcript k using only its friends, the adjusted effective length for friend k' of transcript k can then be estimated by $\widehat{UL}_{k'} + \widehat{ML}_{kk'} = (TL_{k'} - \sum_{k^* \in G_{k'}} \widehat{ML}_{k'k^*}) + \widehat{ML}_{kk'}$. This approach to reducing the size of multiread networks allows for computationally manageable parameter space dimensions when optimizing q-likelihoods.

4.3 Simulation Study

We do not currently have a real data set that includes the multiread counts, $m_{ikk'}$. The simulations here are based on the fly embryo data set described in Chapter 3 and an unpublished maize data set, described below.

We examined an unpublished maize B73 data provided from E. Takacs and M. Scanlon for an experiment comparing gene expression between several domains of developing embryos and fourteen day old seedlings. Two samples were isolated via laser microdisection (LCM) from each of the following six domains in a completely randomized experimental design: proembryo, transition, coleoptile, first leaf, fourteen day old seedlings and lateral meristems from fourteen day old seedlings. After LCM, RNA was isolated from each sample and amplified via poly-A priming and RNA-Polymerase. The amplified RNA was used to generate cDNA libraries and 44-basepair reads were generated using Illumina RNA-sequencing. The RNA sequencing reads generated were aligned to the maize gene space (Harper et al., 2011) and those reads that aligned to the gene space unambiguously were tabulated for each gene. In total, there were 66139 genes with at least one unambiguous read. Further details regarding this experiment are available at

`http://dev.maizegdb.org/cgi-bin/termdoclist.cgi?ref=9021713&type=32466`

on the Maize Genetics and Genomics Database (Harper et al., 2011).

One of the researchers' primary interests was to identify genes that were DE between the fourteen day old seedlings (Seed14) samples and the lateral meristems from fourteen day old seedlings (LM14) samples. The simulations presented in this paper were based on the set of 42204 genes that had at least five reads summing over the four Seed14 and LM14 samples.

4.3.1 Simulation Descriptions

To examine the effectiveness of our suggested approach, we conducted a series of simulations with total sample sizes of 4 and 10, split evenly between two treatment groups. In each simulation, library size factors were simulated according to $\log_2 c_i \sim \text{Normal}(0, .5^2)$, where c_i is the simulated library size factor for the ith sample. Simulated transcripts with average counts less than 1 or more than 500,000 total counts were replaced with new simulated data before analyzing. The former are transcripts whose count data contain little or no information about differential expression that can be detected with any method. The latter represents genes with severely high counts. Each simulation scenario was repeated 50 times, and each data set contained simulated uniread counts for 1500 DE and 3500 EE transcripts. Except where otherwise noted, multiread counts were simulated for 6000

transcript pairs and represented roughly 30% of the total simulated read counts. As a more extreme example, we include a scenario where multireads make up roughly 50% of the total simulated read counts.

We simulated data from gamma-Poisson and discrete gamma models using parameters determined by sample averages and dispersion estimates from the fly embryo and maize data sets, respectively. For the fly embryo simulations, let $\bar{u}_{1k}$ and $\bar{u}_{2k}$ be the sample averages of the two observed uniread counts from the kth gene for the A and B conditions, respectively, and let $\bar{u}_{\cdot k} = (\bar{u}_{1k} + \bar{u}_{2k})/2$. Let $\bar{\Phi}_{1k}$ and $\bar{\Phi}_{2k}$ be the non-shrunken dispersion estimate for the kth gene, based on uniread observations from the two samples in the A and B conditions, respectively (leaving 1 degree of freedom for each). Let $\bar{\Phi}_{\cdot k} = (\bar{\Phi}_{1k} + \bar{\Phi}_{2k})/2$.

Under the gamma-Poisson model, uniread counts for the kth transcript were generated in the following manner. Let k' index a transcript randomly selected from the fly embryo data set. Let $\tilde{\Phi}_{\cdot k'}$ and $\tilde{\Phi}_{tk'}$ be $\max(1.01, \bar{\Phi}_{\cdot k'})$ and $\max(1.01, \bar{\Phi}_{tk'})$, respectively, for $t = 1,2$. If the kth simulated gene was to be EE, we let $\kappa_{tk} = 1/(\tilde{\Phi}_{\cdot k'} - 1)$ and $\alpha_{tk} = \bar{u}_{\cdot k'}(\tilde{\Phi}_{\cdot k'} - 1)$ for $t = 1,2$. If the kth simulated transcript was to be DE, we restricted k' such that $|\bar{u}_{1k'} - \bar{u}_{2k'}| > 4$ and let $\kappa_{tk} = 1/(\tilde{\Phi}_{tk'} - 1)$ and $\alpha_{tk} = \bar{u}_{tk'}(\tilde{\Phi}_{tk'} - 1)$ for $t = 1,2$. We then generated $\mu_{ik}|c_i \sim \text{gamma}(c_i \alpha_{\tau(i)k}, \kappa_{\tau(i)k})$ and, ultimately, $u'_{ik}|\mu_{ik} \sim \text{Poisson}(\mu_{ik})$.

For the maize data simulations, let $\bar{u}_{1k}$ and $\bar{u}_{2k}$ be the sample averages of the two observed uniread counts from the kth gene for the Seed14 and LM14 conditions, respectively, and let $\bar{u}_{\cdot k} = (\bar{u}_{1k} + \bar{u}_{2k})/2$. Let $\hat{\Phi}'_{1k}$ and $\hat{\Phi}'_{2k}$ be the non-shrunken dispersion estimate for the kth gene, based on observations from the two samples in the Seed14 and LM14 conditions, respectively (leaving 1 degree of freedom for each). To reduce variability, we replace $\hat{\Phi}'_{ik}$ with $\hat{\Phi}_{ik} = .3\hat{\Phi}'_{ik} + .7\hat{\Phi}'_{\cdot k}$ for $i = 1,2$, where $\hat{\Phi}'_{\cdot k}$ is the dispersion parameter estimate for gene k based on all 12 samples (with 6 degrees of freedom).

For the discrete gamma model, we simulated uniread data for the kth transcript in the following manner. Let k' index a transcript randomly selected from the maize data set. If the kth simulated transcript was to be EE, we let $\kappa_{tk} = 1/\hat{\Phi}_{\cdot k'}$ (or 1, whichever was smaller) and $\alpha_{tk} = \bar{u}_{\cdot k'}/\hat{\Phi}_{\cdot k'}$ for $t = 1,2$. If the kth simulated gene was to be DE, we restricted k' such that $|\bar{u}_{1k'} - \bar{u}_{2k'}| > 4$ and let $\kappa_{tk} = 1/\hat{\Phi}_{tk'}$ (or 1, whichever was smaller) and $\alpha_{tk} = \bar{u}_{tk'}/\hat{\Phi}_{tk'}$ for $t = 1,2$. We then generated $u'_{ik}|c_i \sim \text{gamma}(c_i \alpha_{\tau(i)k}, \kappa_{\tau(i)k})$. Final simulated uniread counts were taken as u'_{ik} rounded to the nearest integer.

We simulated multireads counts using the following procedure. For each simulated transcript, an effective length for unireads UL_k was simulated for each transcript from a lognormal(8.2,1) distribution truncated above at 15000. A pair of simulated transcript indexes, k and k', were randomly chosen from the set of integers from 1 to 5000, and a multiread effective length $ML_{kk'}$ was simulated from a lognormal(5.5,1) distribution truncated above and below at 1500 and 10, respectively. The total effective length for each simulated transcript was recorded as $TL_k = UL_k + \sum_{k' \in G_k} ML_{kk'}$ and treated as fixed and known during analysis. For simulations

from the discrete gamma model, multiread counts were then simulated as $m_{ikk'} = \sum_{p=k,k'} \text{round}(m^*_{ip})$ where $m^*_{ip} \sim \text{gamma}(c_i ML_{kk'}/UL_p \alpha_{\tau(i)p}, \kappa_{\tau(i)p})$. For simulations from the gamma-Poisson model, multiread counts were then simulated as $m_{ikk'} = \sum_{p=k,k'} m^*_{ip}$), where $m^*_{ip} \sim \text{Poisson}(\mu^*_{ip})$ and $\mu^*_{ip}|c_i \sim \text{gamma}(c_i ML_{kk'}/UL_p \alpha_{\tau(i)k}, \kappa_{\tau(i)p})$.

Under the assumption of uniformly distributed reads, the quasi-Poisson model can be written as

$$u_{ik}|UL_k, TL_k, \mu_{ik} \sim \text{ quasi-Poisson}((1 - \sum_{k' \in G_k} \pi_{kk'})\mu_{ik}) \tag{4.13}$$

and

$$m_{ikk'}|ML_{kk'}, TL_k, TL_{k'}, \mu_{ik}, \mu_{ik'} \sim \text{ quasi-Poisson}(\pi_{kk'}\mu_{ik} + \pi_{k'k}\mu_{ik'}), \tag{4.14}$$

where $\pi_{kk'} = ML_{kk'}/TL_k$ is the proportion of reads from transcript k that will map to both transcripts k and k'. Estimating values for $ML_{kk'}$ from the data when TL_k and $TL_{k'}$ are fixed and known is equivalent to estimating proportions $\pi_{kk'}$ subject to the restrictions that $\pi_{kk'} \geq 0$, $\sum_{k' \in G_k} \pi_{kk'} \leq 1$ and $\pi_{kk'} TL_k = \pi_{k'k} TL_{k'}$ $\forall (k,k') \in S^2_k$.

Many experiments have shown that reads within a transcript are not uniformly distributed (i.e. that each possible read sequence within a transcript does not have the same chance of being observed). The distribution of reads within a transcript is subject to positional bias (Bohnert and R$^{\acute{}}$atsch, 2010; Howard and Heber, 2010; Li et al., 2010a; Wu et al., 2010), sequence bias (Hansen et al., 2010; Li et al., 2010b; Turro et al., 2011), and biases of unknown sources (Li et al., 2010b; Roberts et al., 2011). Therefore, we also examined the performance of our suggested method when the uniform read distribution assumption is violated. To violate this assumption, we conducted simulations of unireads and total effective lengths exactly as described above. Before simulating count data for the multireads shared between transcripts k and k', we simulated two perturbation variables δ_k and $\delta_{k'}$ independently from a lognormal(0,.2) distribution.

For simulations from the discrete gamma model, multiread counts were then simulated as

$$m_{ikk'} = \sum_{p=k,k'} \text{round}(m^*_{ip}),$$

where

$$m^*_{ip} \sim \text{gamma}(c_i \delta_p ML_{kk'}/UL_p \alpha_{\tau(i)p}, \kappa_{\tau(i)p}).$$

For simulations from the gamma-Poisson model, multiread counts were then simulated as

$$m_{ikk'} = \sum_{p=k,k'} m^*_{ip},$$

where

$$m^*_{ip} \sim \text{Poisson}(\mu^*_{ip})$$

and

$$\mu^*_{ip}|c_i \sim \text{gamma}(c_i \delta_p ML_{kk'}/UL_p \alpha_{\tau(i)k}, \kappa_{\tau(i)p}).$$

The model for this simulation approach can be written as

$$u_{ik}|UL_k, TL_k, \mu_{ik} \sim \text{quasi-Poisson}((1 - \sum_{k' \in G_k} \pi_{kk'})\mu_{ik})$$

and

$$m_{ikk'}|ML_{kk'}, TL_k, TL_{k'}, \mu_{ik}, \mu_{ik'} \sim \text{quasi-Poisson}(\pi_{kk'}\mu_{ik} + \pi_{k'k}\mu_{ik'}),$$

where $\pi_{kk'}TL_k \neq \pi_{k'k}TL_{k'} \forall (k,k') \in S_k^2$, thus violating the constraints of the assumption of uniformly mapped reads.

4.3.2 Simulation Results

Each simulated data set was analyzed with the QL and QLSpline methods applied to the unireads and separately to the combination of uniread and multiread networks formed by including each transcript and its friends using the methods described in Section 4.2. For each complete multiread network, $ML_{kk'}$ was estimated $\forall (k,k') \in S_k^2$ using the suggested iterative method. The resulting estimates $\widehat{ML}_{kk'}$ were considered fixed and known and along with the true values of TL_k were used to set $\hat{\pi}_{kk'} = \widehat{ML}_{kk'}/TL_k$. Allowing the values of $\hat{\pi}_{kk'}$ to be chosen by optimizing the q-likelihood with the β_k parameters produced similar results as when we treated the $\hat{\pi}_{kk'}$ values from the iterative procedure as fixed and known. We therefore chose to fix the estimates $\hat{\pi}_{kk'}$ to their values from the iterative procedure. For each transcript, we used the q-likelihood formed by using unireads from the transcript itself and its friends in addition to multireads involving the target transcript. Under this approach, optimizing the q-likelihood function only involves choosing values for β_k and $\beta_{k'} \forall k' \in G_k$. We evaluated each method's performance according to two criteria: the separation of DE and EE transcipts in significance rankings as seen in receiver operatoring characteristic (ROC) curves and the accuracy of estimated false discovery rates (FDR) as observed by comparing empirical FDRs (eFDR) to q-values (Storey and Tibshirani, 2003).

We examined the significance rankings of each method using ROC curves. The solid curves in Figures 4.1 and 4.2 display ROC curves from the fly embryo and maize scenarios, respectively, averaged across fifty simulations. Figure 4.5 displays ROC curves from $n = 4$ maize simulations where multireads made up roughly half of all reads. The solid thin lines are $\pm$ two standard errors around the mean, providing approximate 95% pointwise confidence intervals. These plots show that incorporating multireads improves the significance rankings for both the QL and QLSpline methods. The observed improvement is most pronounce for simulations where $n = 4$.

To facilitate direct comparison between the methods, we summarized the significance rankings of each method by examining the average sensitivity at a specificity of 0.99. Tables 4.2 and 4.3 provide the average proportion of DE genes that have p-values below that of the 35th most significant EE transcript from the $n = 4$ simulations based on the fly embryo and maize data sets, respectively. Paired t-tests revealed that the average sensitivity of the MultQLSpline method was significantly greater than that of each alternative method in each of nine

simulation scenarios. (The corresponding two-sided p-values were all less than 0.01.) In the maize simulation scenario where multireads made up roughly half of the total reads, the average sensitivities for the MultQLSpline method and its closest competitor, QLSpline, method were 0.386 and 0.322, respectively. Overall, these results demonstrate that the MultQLSpline method, on average, produced better significance rankings than any other method across these simulations.

The average number of truly DE transcripts contained in lists of the 250 (500) most significant transcripts for the $n = 4$ ($n = 10$) simulations from the fly embryo and maize data sets are provided in Tables 4.2 and 4.3, respectively. Although the methods that incorporate multireads produced improved significance rankings, there was little difference between the uniread-only and multiread methods in these averages for these simulations. For the simulation scenario in which roughly half of all reads were multireads, the MultQLSpline method had an average of 247.2 truly DE transcripts in its lists of 250 most significant transcripts, while its nearest competitor, QLSpline, averaged 244.4.

We examined the conservative or liberal nature of each method by comparing eFDRs to q-values. Each transcript's eFDR reported the proportion of transcripts that were EE from the set of transcripts that had p-values as small as or smaller than that of the transcript itself. Q-values were obtained by applying the method of Nettleton et al. (2006) to the distribution of p-values resulting from the application of each method. If a method is neither conservative nor liberal, then the q-value for any given transcript should closely match its eFDR. For example, if the transcript with the Wth smallest p-value has a corresponding q-value of .05, then roughly 5% of the W transcripts with p-values as small or smaller should be EE.

To examine if this characteristic holds for each method, we plotted average eFDRs versus q-values for each scenario. The solid curves in Figures 4.3 and 4.4 display curves from the fly embryo and maize scenarios, respectively, averaged across fifty simulations. Figure 4.5 displays eFDR curves from $n = 4$ maize simulations where multireads made up roughly half of all reads. The solid thin lines are $\pm$ two standard errors around the mean, providing approximate 95% confidence intervals. To construct these plots, we rounded each q-value to the nearest 0.001 before plotting. When multiple transcripts produced identical rounded q-values for a given method, the eFDR of the transcript with the largest original p-value was used to represent the set. (This technique facilitated averaging of eFDRs across simulations and computating standard errors at each rounded q-value.) If a method was neither conservative nor liberal, its line should closely follow the brown $y = x$ diagonal. Lines appearing substantially above (below) the brown diagonal indicate the corresponding method was liberal (conservative).

For each $n = 10$ simulation scenario in our study, the average eFDR curves for the QL and MultQL methods are substantially above the brown $y = x$ diagonal, indicating these methods produced liberal results for these data. The plots for $n = 4$ simulation scenarios do not show lines for the QL or MultQL methods, as they were

so conservative that few, if any, transcripts had q-values less than 0.1. The QLSpline was conservative in each simulation scenario. The MultQLSpline was slightly conservative (less so than the QLSpline method), or accurate in all scenarios but one. In $n = 4$ simulations from the maize data set when multireads represented roughly 50% of all reads, the MultQLSpline methods was slightly liberal. The average eFDR with a corresponding q-value of 0.05 for each method are provided as numerical summaries for the fly embryo and maize data simulations in Tables 4.2 and 4.3, respectively.

4.4 Discussion

We have suggested a method for modeling multireads and demonstrated that doing so leads to improved significance rankings. There is much opportunity for further extension to these methods. We have estimated dispersion parameters for each transcript using only its uniread counts. When modeling multireads, we have assumed that the target transcript and its included friends share a common dispersion parameter.

The methods described in this chapter were developed assuming that each sequence of R bases from within a transcript is equally likely to appear as a read. Many factors such as positional bias (Bohnert and R$'$atsch, 2010; Howard and Heber, 2010; Li et al., 2010a; Wu et al., 2010), sequence bias (Hansen et al., 2010; Li et al., 2010b; Turro et al., 2011), and biases of unknown sources (Li et al., 2010b; Roberts et al., 2011) can influence the read appearance rates of sequences within a transcript. Although doing so may drastically increase the computational cost of their implementation, the methods described above can easily accommodate such influences when the factors and their corresponding biases are well understood. The incorporation of multireads as implemented in the methods described in this chapter only requires (an estimate of) the proportions of reads occurring from each transcript that would be uniquely mapped to the transcript itself and that would be multireads separately for each multiread "partner".

Perhaps most important, the methods could be generalized to handle mean-variance relationships other than $\text{Var}(Y)$=$\Phi\text{E}(Y)$. A multiread method based on the negative-binomial distribution would be of special interest given the multitude of existing RNA-seq methods based on this distribution. However, a linear mean-variance relationship is convenient for directly modeling uniread counts and multiread counts, rather than modeling the latent sum of all reads originating from each transcript in each sample.

Table 4.1 Legend for simulation numerical summaries.

Statistic Labels	
$Sens_{Spec=.99}$	Sensitivity when specificity=.99
# DE Top QQQ	Number of truly DE genes contained in list of QQQ most significant genes
eFDR	empirical FDR for list of all genes with q-values less than .05
$N_{Q<.05}$	Number of genes with q-values less than .05
$\hat{P}_{EE}$	Estimated proportion of null genes
Max SE	Maximum standard error of averages in each row

Significance Markings	
$^{\circ}$	paired t-test comparing reported average to that of MultQL yielded two-sided p-value$<$0.01
*	paired t-test comparing reported average to that of MultQLSpline yielded two-sided p-value$<$0.01

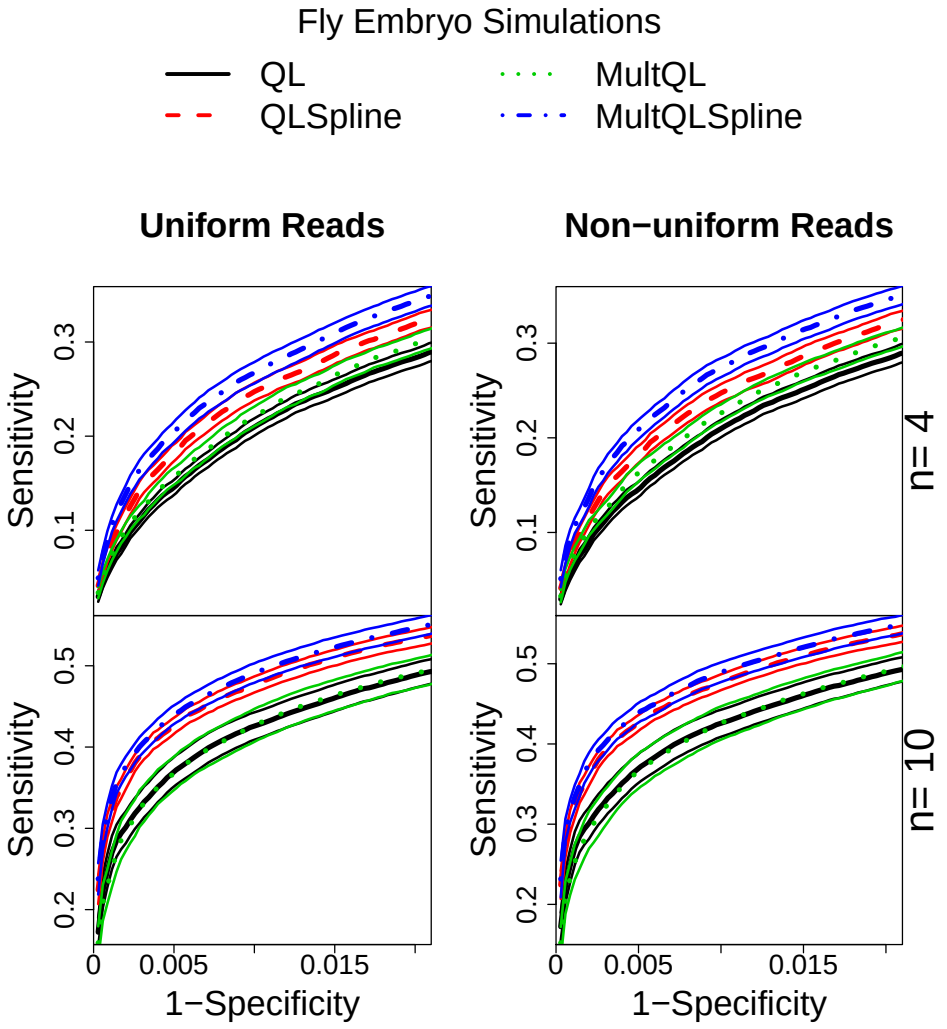

Figure 4.1 Average ROC curves for fly embryo data simulations from gamma-Poisson model using uniform (left) and non-uniform (right) read distributions with $n = 4$ (top) and $n = 10$ (bottom).

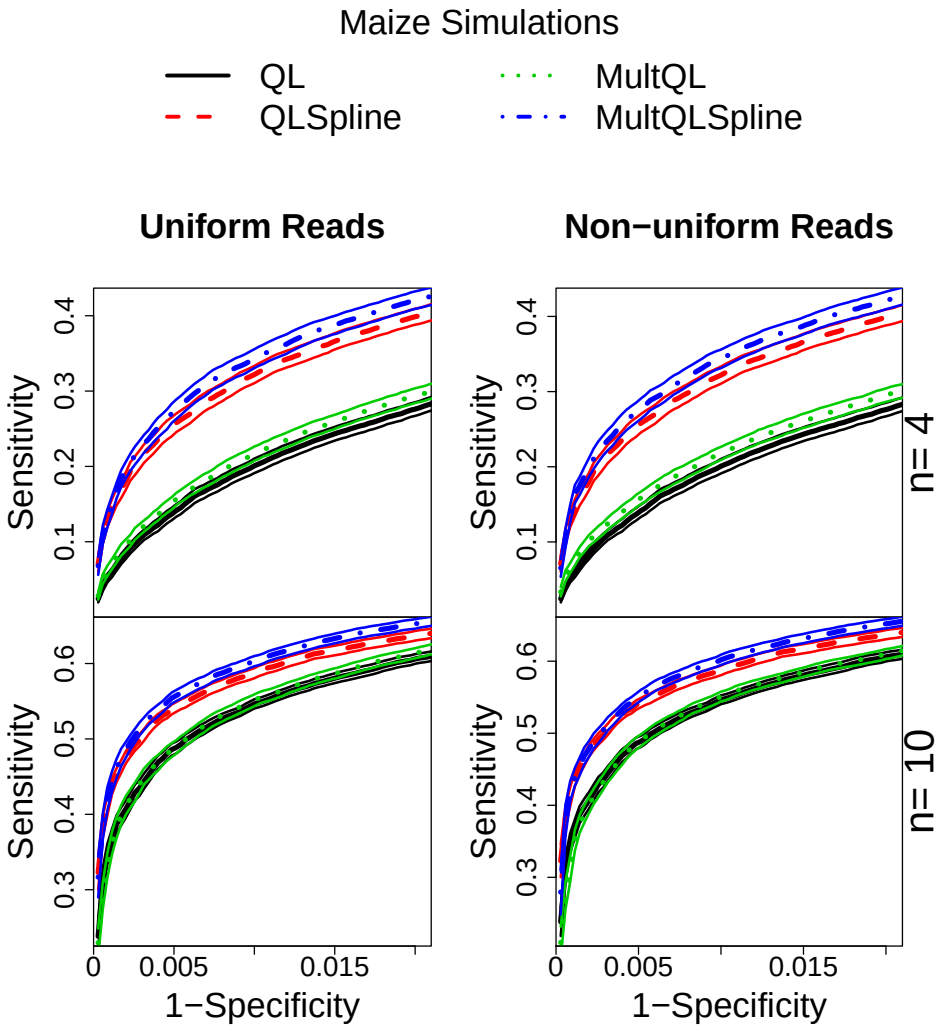

Figure 4.2 Average ROC curves for maize data simulations from discrete gamma model using uniform (left) and non-uniform (right) read distributions with $n = 4$ (top) and $n = 10$ (bottom).

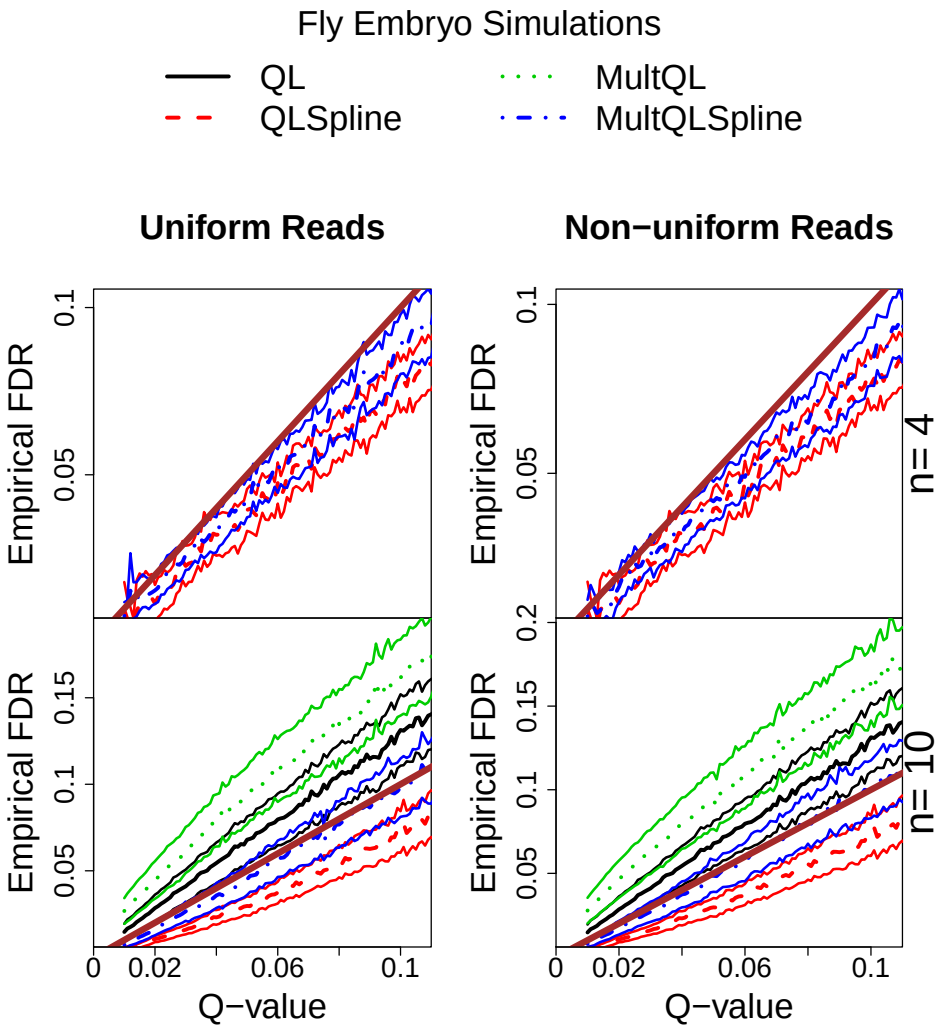

Figure 4.3 Curves relating average eFDR to q-values for fly embryo data simulations from gamma-Poisson model using uniform (left) and non-uniform (right) read distributions with $n = 4$ (top) and $n = 10$ (bottom).

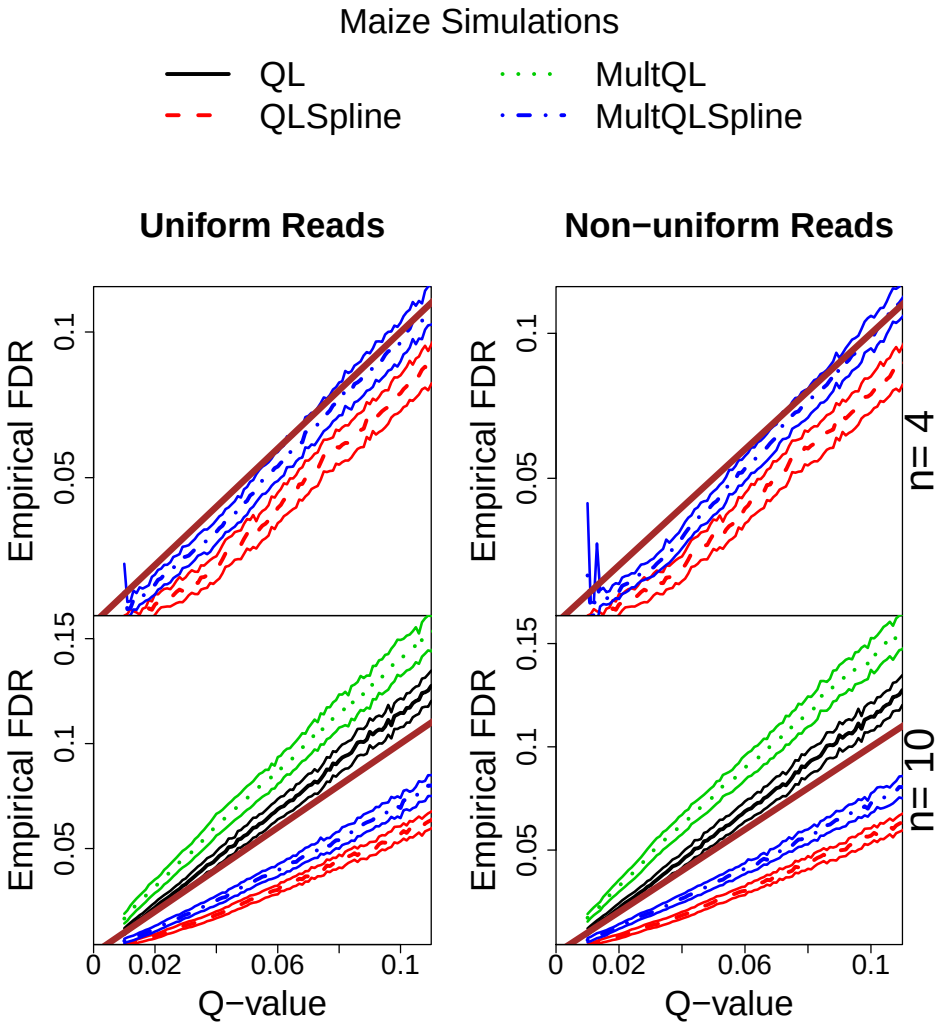

Figure 4.4 Curves relating average eFDR to q-values for maize data simulations from discrete gamma model using uniform (left) and non-uniform (right) read distributions with $n = 4$ (top) and $n = 10$ (bottom).

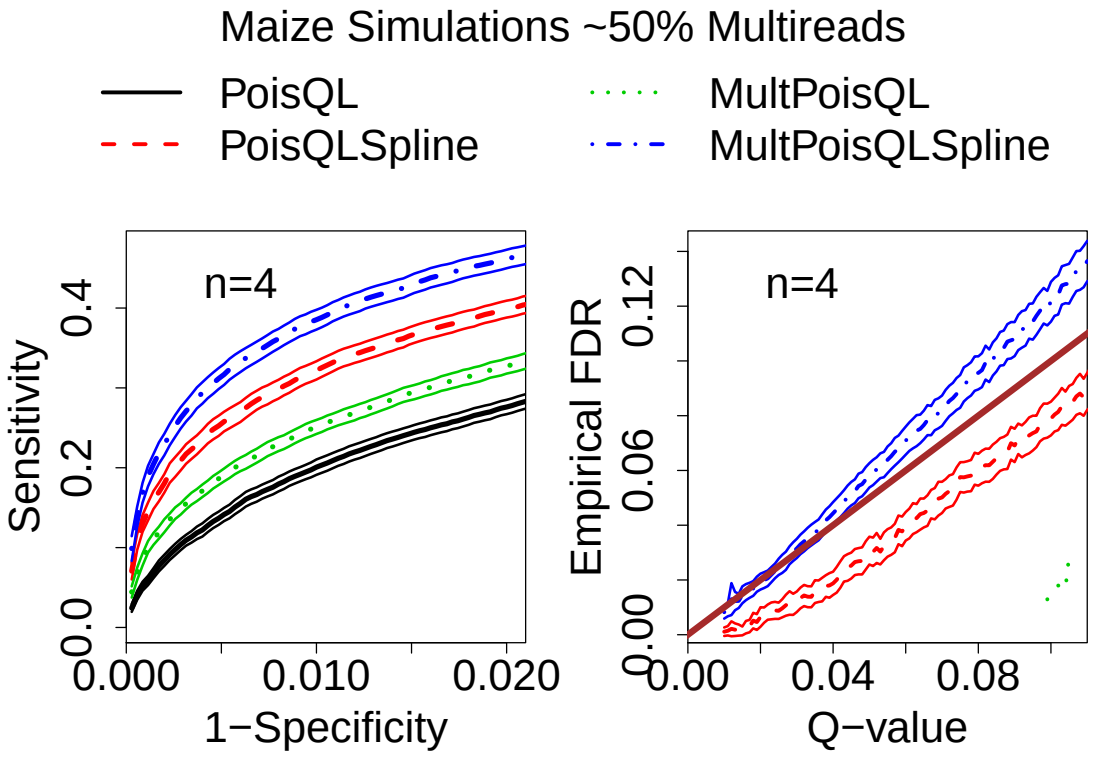

Figure 4.5 ROC curves (left) and curves relating average eFDR to q-values (right) for $n = 4$ maize data simulations from discrete gamma model when multireads make up 50% of all reads.

Table 4.2 Summary of results from discrete gamma simulations based on fly embryo data set. See Table 4.1 for legend.

	QL	QLSpline	MultQL	MultQLSpline	Max SE
$n = 4$ Fly GP Uniform Read Distribution					
$Sens_{Spec=.99}$	$0.21^{\circ\star}$	$0.247^{\circ\star}$	$0.221^{\star}$	0.267°	0.0057
# DE Top 250	$230.6^{\circ\star}$	$237.1^{\circ\star}$	232.2	239.9°	0.92
eFDR	0	0.0393	0	0.043	0.00313
$N_{Q<.05}$	0	181.6	0	267.8	12.5
$\hat{P}_{EE}$	0.893	0.801	0.871	0.771	0.00673
$n = 4$ Fly GP Non-uniform Read Distribution					
$Sens_{Spec=.99}$	$0.21^{\circ\star}$	$0.247^{\circ\star}$	$0.226^{\star}$	0.274°	0.00499
# DE Top 250	230.6°	$237.1^{\star}$	$233.9^{\star}$	240.3°	0.81
eFDR	0	0.0393	0	0.0418	0.00313
$N_{Q<.05}$	0	181.6	0	271.9	12.7
$\hat{P}_{EE}$	0.893	0.801	0.865	0.771	0.00689
$n = 10$ Fly GP Uniform Read Distribution					
$Sens_{Spec=.99}$	$0.426^{\star}$	$0.477^{\circ\star}$	$0.427^{\star}$	0.491°	0.0101
# DE Top 500	$488.2^{\star}$	$495.6^{\circ\star}$	487.3	496.5	1.39
eFDR	0.068	0.0297	0.0934	0.045	0.00829
$N_{Q<.05}$	715.9	668.8	822.4	755.8	15.6
$\hat{P}_{EE}$	0.723	0.77	0.698	0.743	0.00903
$n = 10$ Fly GP Non-uniform Read Distribution					
$Sens_{Spec=.99}$	$0.426^{\star}$	$0.477^{\circ\star}$	$0.426^{\star}$	0.489°	0.0104
# DE Top 500	$488.2^{\star}$	495.6°	486.8	496.1	1.43
eFDR	0.068	0.0297	0.0934	0.0488	0.00845
$N_{Q<.05}$	715.9	668.8	825.6	762.1	16.2
$\hat{P}_{EE}$	0.723	0.77	0.696	0.739	0.00994

Table 4.3 Summary of results from gamma-Poisson simulations based on maize data set. See Table 4.1 for legend.

	QL	QLSpline	MultQL	MultQLSpline	Max SE
$n = 4$ Maize DG Uniform Read Distribution					
$Sens_{Spec=.99}$	$0.201^{\circ\star}$	$0.322^{\circ\star}$	$0.217^{\star}$	0.343°	0.00627
# DE Top 250	$229.1^{\circ\star}$	$244.4^{\circ\star}$	$232.3^{\star}$	245.3°	0.9
eFDR	0	0.0308	0	0.0419	0.00248
$N_{Q<.05}$	0	313.7	0	444.2	17.7
$\hat{P}_{EE}$	0.784	0.775	0.76	0.743	0.00433
$n = 4$ Maize DG Non-uniform Read Distribution					
$Sens_{Spec=.99}$	$0.201^{\circ\star}$	$0.322^{\circ\star}$	$0.218^{\star}$	0.344°	0.00617
# DE Top 250	$229.1^{\circ\star}$	244.4°	$232.4^{\star}$	245.4°	0.9
eFDR	0	0.0308	0	0.0414	0.00248
$N_{Q<.05}$	0	313.7	0	439.7	17.7
$\hat{P}_{EE}$	0.784	0.775	0.751	0.746	0.00524
$n = 10$ Maize DG Uniform Read Distribution					
$Sens_{Spec=.99}$	$0.546^{\star}$	$0.587^{\circ\star}$	$0.551^{\star}$	0.603°	0.00391
# DE Top 500	$497.7^{\star}$	499.3°	$497.2^{\star}$	499.2°	0.27
eFDR	0.0567	0.0247	0.0741	0.0329	0.00282
$N_{Q<.05}$	926.9	855	1005.9	922.2	8.9
$\hat{P}_{EE}$	0.708	0.788	0.685	0.766	0.00478
$n = 10$ Maize DG Non-uniform Read Distribution					
$Sens_{Spec=.99}$	$0.546^{\star}$	$0.587^{\circ\star}$	$0.551^{\star}$	0.602°	0.00356
# DE Top 500	$497.7^{\circ\star}$	499.3°	$497^{\star}$	499°	0.27
eFDR	0.0567	0.0247	0.076	0.0338	0.00248
$N_{Q<.05}$	926.9	855	1008.5	923.2	8.6
$\hat{P}_{EE}$	0.708	0.788	0.684	0.765	0.00494
$n = 4$ Maize DG Uniform Read Dist. with 50% Multireads					
$Sens_{Spec=.99}$	$0.201^{\circ\star}$	$0.322^{\circ\star}$	$0.251^{\star}$	0.386°	0.00619
# DE Top 250	$229.1^{\circ\star}$	$244.4^{\circ\star}$	$238.3^{\star}$	247.2°	0.9
eFDR	0	0.0308	0	0.0581	0.00248
$N_{Q<.05}$	0	313.7	0	619.4	17.7
$\hat{P}_{EE}$	0.784	0.775	0.723	0.715	0.00468

CHAPTER 5. Conclusion

The identification of differentially expressed genes is an important tool used in the quest to understand gene function. Microarray and RNA-seq experiments provide the expression data for thousands of genes. The analysis of expression data is a popular and rapidly developing area of research. While many methods for detecting differentially expressed genes exist, several important contributions have been made by the works presented in this dissertation. For both microarray and RNA-seq data, we have developed methods that allow gene-specific parameter estimators and account for their corresponding uncertainty. We have also described an approach for modeling multireads (data that is often discarded) and have shown that doing so can improve the detection of differentially expressed genes or transcripts. Our suggested methods have been shown to improve the ability to sort differentially expressed genes from equivalently expressed genes and often to improve the control of false discovery rates.

In Chapter 2, we have discussed empirical Bayes methods for modeling microarray data. Many methods for microarray analysis have been developed under the assumption that differences within a gene due to changes in environment can be modeled in the same manner as differences across genes. We have shown this assumption to be inaccurate and harmful to the detection of differential expression. We have demonstrated how such an assumption can be relaxed, producing methods with improved power. We have also shown that methods assuming that error variances are constant across genes or that gene-specific variance estimators have no uncertainty produce inaccurate posterior probabilities of differential expression. The liberal inaccuracies of these methods posterior probabilities can cause researchers who use them to underestimate the proportion of false positives on a list of potentially differentially expressed genes. We have suggested methods for accounting for the uncertainty in gene-specific variance estimators and demonstrated that it produces accurate posterior probabilities of differential expression.

In Chapter 3, we have focused on identifying differentially expressed genes from RNA-seq data. We have presented a general use quasi-likelihood approach and demonstrated its use for quasi-Poisson and quasi-negative binomial models. Quasi-likelihood models can provide more flexible mean-variance relationships compared to fully specified models. Our suggested hypothesis tests for differential expression are analogous to those used in standard ANOVA analyses, with deviance playing as an analog to sums of squares. We have shown that our

suggested tests based on a quasi-negative binomial model offer several advantages over other popular approaches based on negative binomial models, including improved error rate control and greater robustness to model mis-specification.

The approach discussed in Chapter 4 offers promise as a way to retain RNA-seq data for reads that align with multiple reference genes or transcripts when analyzing for differential expression. This approach has been shown to improve significance rankings for RNA sequencing experiments with many multireads. Future work in this area may provide more flexible models (beyond quasi-Poisson) and improved control of false discovery rates.

Bibliography

Anders, S. and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome Biology*, 11.

Auer, P. L. and Doerge, R. W. (2011). A two-stage poisson model for testing RNAseq data. *Statistical Applications in Genetics and Molecular Biology*, 10(1).

Baldi, P. and Long, A. D. (2001). A Bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes. *Bioinformatics*, 17(6):509–519.

Binder, H., Kirsten, T., Loeffler, M., and Stadle, P. F. (2004). Sensitivity of microarray oligonucleotide probes: Variability and effect of base composition. *The Journal of Physical Chemistry B*, 108(46):18003–18014.

Blekhman, R., Marioni, J. C., Zumbo, P., Stephens, M., and Gilad, Y. (2010). Sex-specific and lineage-specific alternative splicing in primates. *Genome Research*, 20:180–189.

Bohnert, R. and R´atsch, G. (2010). rquant.web: a tool for RNA-seq-based transcript quantitation. *Nucleic Acids Research*, 38:W348–W351.

Bullard, J. H., Purdom, E., Hansen, K. D., and Dudoit, S. (2010). Evaluation of statistical methods for normalization and differential expression in mRNA-seq experiments. *BMC Bioinformatics*, 11(94).

Cui, X., Hwang, J. T. G., Qiu, J., Blades, N. J., and Churchill, G. A. (2005). Improved statistical tests for differential gene expression by shrinking variance components. *Biostatistics*, 6(1):59–75.

Cumbie, J. S., Kimbrel, J. A., Di, Y., Schafer, D. W., Wilhelm, L. J., Fox, S. E., Sullivan, C. M., Curzon, A. D., Carrington, J. C., Mockler, T. C., and Chang, J. H. (2011). GENE-counter: A computational pipeline for the analysis of RNA-seq data for gene expression differences. *PLoS ONE*, 6(10).

Di, Y., Schafer, D. W., Cumbie, J. S., and Chang, J. H. (2011). The NBP negative binomial model for assessing differential gene expression from RNA-seq. *Statistical Applications in Genetics and Molecular Biology*, 10(1).

Hansen, K. D., Brenner, S. E., and Dudoit, S. (2010). Biases in Illumina transcriptome sequencing caused by random hexamer priming. *Nucleic Acids Research*, 38.

Harper, L. C., Schaeffer, M. L., Thistle, J., Gardiner, J. M., Andorf, C. M., Campbell, D. A., Cannon, E. K., Braun, B. L., Birkett, S. M., Lawrence, C. J., and Sen, T. Z. (2011). The MaizeGDB Genome Browser tutorial: one example of database outreach to biologists via video. *Database*.

Howard, B. E. and Heber, S. (2010). Towards reliable isoform quantification using RNA-seq data. *BMC Bioinformatics*, 11. Suppl 3.

Irizarry, R. A., Hobbs, B., Collin, F., Beazer-Barclay, Y. D., Antonellis, K. J., Scherf, U., and Speed, T. P. (2003). Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4(2):249–264.

Jensen, S. T., Erkan, I., Arnardottir, E. S., and Small, D. S. (2009). Bayesian testing of many hypotheses x many genes: A study of sleep apnea. *Annals of Applied Statistics*, 3(3):1080–1101.

Keles, S. (2007). Mixture modeling for genome-wide localization of transcription factors. *Biometrics*, 63(1):10–21.

Kendziorski, C. M., Newton, M., Lan, H., and Gould, M. N. (2003). On parametric empirical Bayes methods for comparing multiple groups using replicated gene expression profiles. *Statistics in Medicine*, 22(1):3899–3914.

Li, B., Ruotti, V., Stewart, R. M., Thomson, J. A., and Dewey, C. N. (2010a). RNA-seq gene expression estimation with read mapping uncertainty. *Bioinformatics*, 26(4):493–500.

Li, J., Jiang, H., and Wong, W. H. (2010b). Modeling non-uniformity in short-read rates in RNA-seq data. *Genome Biology*, 11.

Lo, K. and Gottardo, R. (2007). Flexible empirical Bayes models for differential gene expression. *Bioinformatics*, 23(3):328–335.

Lonnstedt, I. and Speed, T. P. (2002). Replicated microarray data. *Statistica Sinica*, 12(1):31–46.

Lu, J., Tomfohr, J. K., and Kepler, T. B. (2005). Identifying differential expression in multiple SAGE libraries: an overdispersed log-linear model approach. *Bioinformatics*, 6(165).

Marioni, J. C., Mason, C. E., Mane, S. M., Stephens, M., and Gilad, Y. (2008). RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays. *Genome Research*, 18:1509–1517.

McCarthy, D. J., Chen, Y., and Smyth, G. K. (2012). Differential expression analysis of multifactor RNA-seq experiments with respect to biological variation. *Nucleic Acids Research*. Online publication 28 January 2010.

McCullagh, P. (1983). Quasi-likelihood functions. *Annals of Statistics*, 11(1):59–67.

McCullagh, P. and Nelder, J. A. (1983). *Generalized Linear Models*. Chapman and Hall, New York, first edition.

Nettleton, D., Hwang, J. T. G., Caldo, R. A., and Wise, R. P. (2006). Estimating the number of true null hypotheses from a histogram of p-values. *Journal of Agricultural, Biological, and Environmental Statistics*, 11(3):337–356.

Newton, M. A., Noueiry, A., Sarkar, D., and Ahlquist, P. (2004). Detecting differential gene expression with a semiparametric hierarchical mixture method. *Biostatistics*, 5(2):155–176.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Roberts, A., Trapnell, C., Donaghey, J., Rinn, J. L., and Pachter, L. (2011). Improving RNA-seq expression estimates by correcting for fragment bias. *Genome Biology*, 12.

Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140.

Robinson, M. D. and Smyth, G. K. (2007). Small-sample estimation of negative binomial dispersion, with applications to SAGE data. *Biostatistics*, 9(2):321–332.

Robinson, M. D. and Smyth, G. K. (2008). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23(21):2881–2887.

Rossell, D. (2009). Gaga: A parsimonious and flexible model for differential expression analysis. *Annals of Applied Statistics*, 3(3):1035–1051.

Selinger, D. W., Saxena, R. M., Cheung, K. J., Church, G. M., and Roseno, C. (2003). Global RNA half-life analysis in escherichia coli reveals positional patterns of transcript degradation. *Genome Research*, 13:216–223.

Smyth, G. K. (2004). Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1).

Somel, M., Creely, H., Franz, H., Mueller, U., Lachmann, M., Khaitovich, P., and Pbo, S. (2008). Human and chimpanzee gene expression differences replicated in mice fed different diets. *PLoS One*, 3(1).

Storey, J. D. and Tibshirani, R. (2003). Statistical significance for genome wide studies. *Proceedings of the National Academy of Sciences*, 100(16):9440–9445.

Tjur, T. (1998). Nonlinear regression, quasi likelihood, and overdispersion in generalized linear models. *American Statistician*, 52(3):222–227.

Turro, E., Su, S.-Y., Gonçalves, A., Coin, L. J., Richardson, S., and Lewin, A. (2011). Haplotype and isoform specific expression estimation using multi-mapping RNA-seq reads. *Genome Biology*, 12.

Vêncio, R. Z., Brentani, H., Patrão, D. F., and Pereira, C. A. (2004). Bayesian model accounting for within-class biological variability in serial analysis of gene expression (SAGE). *BMC Bioinformatics*, 5:119–131.

Wei, Z. and Li, H. (2007). Markov random field model for network-based analysis of genomic data. *Bioinformatics*, 23(12):1537–1544.

Wei, Z. and Li, H. (2008). A hidden spatial-temporal Markov random field model for network-based analysis of time course gene expression data. *Annals of Applied Statistics*, 2(1):408–429.

Wright, G. W. and Simon, R. M. (2003). A random variance model for detection of differential gene expression in small microarray experiments. *Bioinformatics*, 19(18):2448–2455.

Wu, H., Yuan, M., Kaech, S. M., and Halloran, M. E. (2007). A statistical analysis of memory cd8 t cell differentiation: An application of a hierarchical state space model to a short time course microarray experiment. *Annals of Applied Statistics*, 1(2):442–458.

Wu, Z., Wang, X., and Zhang, X. (2010). Using non-uniform read distribution models to improve isoform expression inference in RNA-seq. *Bioinformatics*, 27(4):502–508.

Yuan, M. (2006). Flexible temporal expression profile modelling using the gaussian process. *Computational Statistics and Data Analysis*, 51(3):1754–1764.

Yuan, M. and Kendziorski, C. (2006a). Hidden markov models for microarray time course data in multiple biological conditions. *JASA*, 101(476):1323–1332.

Yuan, M. and Kendziorski, C. (2006b). A unified approach for simultaneous gene clustering and differential expression identification. *Biometrics*, 62(4):1089–1098.

Zhou, Y.-H., Xia, K., and Wright, F. A. (2011). A powerful and flexible approach to the analysis of RNA sequence count data. *Bioinformatics*, (Advanced Access).